

Felipe Regis Gouveia Ribeiro

Identificação da Área Representativa da Retinopatia Diabética com Redes Neurais Convolucionais

João Pessoa – PB
Agosto de 2023

Felipe Regis Gouveia Ribeiro

Identificação da Área Representativa da Retinopatia Diabética com Redes Neurais Convolucionais

Dissertação de Mestrado submetida ao Programa de Pós-Graduação em Engenharia Elétrica do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, como requisito necessário à obtenção do grau de Mestre em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração: Processamento de Sinais

Carlos Danilo Miranda Regis, D.Sc.
Orientação

João Pessoa – PB, Agosto de 2023
© Felipe Regis Gouveia Ribeiro — feliper.g.ribeiro@gmail.com

Dados Internacionais de Catalogação na Publicação (CIP)
Biblioteca Nilo Peçanha - *Campus* João Pessoa, PB.

R484i Ribeiro, Felipe Regis Gouveia.

Identificação da área representativa da retinopatia diabética com redes neurais convolucionais / Felipe Regis Gouveia Ribeiro. – 2023.

87 f. : il.

Dissertação (Mestrado em Engenharia elétrica) – Instituto Federal de Educação da Paraíba / Programa de Pós-Graduação em Engenharia Elétrica. (PPGEE), 2023.

Orientação : Profº D.r Carlos Danilo Miranda Regis.

1.Retinopatia diabética. 2. Processamento de imagens. 3. Arquitetura de redes convolucionais. 4. Redimensionamento de imagem. 5. Marcação da área patológica. I.Título.

CDU 617.735:004.932(043)

Elaboração: Lucrecia Camilo de Lima, Bibliotecária – CRB 15/132



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

PROGRAMA DE PÓS-GRADUAÇÃO *STRICTO SENSU*

MESTRADO EM ENGENHARIA ELÉTRICA

FELIPE REGIS GOUVEIA RIBEIRO

**IDENTIFICAÇÃO DA ÁREA REPRESENTATIVA DA RETINOPATIA DIABÉTICA COM REDES NEURAIS
CONVOLUCIONAIS**

Dissertação apresentada como requisito para obtenção do título de Mestre em Engenharia Elétrica, pelo Programa de Pós- Graduação em Engenharia Elétrica do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba – IFPB - Campus João Pessoa.

Aprovado em 03 de agosto de 2023.

Membros da Banca Examinadora:

Dr. Carlos Danilo Miranda Regis

Dr. Leonardo Vidal Batista

UFPB

Documento assinado digitalmente
gov.br WASLON TERLIZZIE ARAUJO LOPES
Data: 28/08/2023 12:04:11-0300
Verifique em <https://validar.it.gov.br>

Dr. Waslon Terlizzie Araújo Lopes

UFPB

Dra. Aganeide Castilho Palitot

UFPB

João Pessoa/2023

Documento assinado eletronicamente por:

- **Carlos Danilo Miranda Regis**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 10/08/2023 10:45:37.
- **Leonardo Vidal Batista**, PROFESSOR DE ENSINO SUPERIOR NA ÁREA DE ORIENTAÇÃO EDUCACIONAL, em 10/08/2023 13:07:46.
- **AGANEIDE CASTILHO PALITOT**, PROFESSOR DE ENSINO SUPERIOR NA ÁREA DE ORIENTAÇÃO EDUCACIONAL, em 14/08/2023 14:52:41.

Este documento foi emitido pelo SUAP em 11/07/2023. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código 448037
Verificador: 114d0cb7e5
Código de Autenticação:



DEDICATÓRIA

Este trabalho é dedicado, em primeiro lugar aos meus pais, Lucia e Sergio, aos meus amados filhos Fernando e Guilherme e a minha esposa Gabriella.

AGRADECIMENTOS

Agradeço primeiramente, a Deus, pelo dom da vida, pelos desafios, pelas conquistas e por ter me permitido ir tão longe.

A toda a minha família que sempre me deu forças no decorrer do curso, aos meus pais, irmão, esposa e filhos, pela compreensão nas ausências necessárias ao curso. Ao meu primo e compadre Thiago Regis pela interminável leitura, sugestões e correções no texto.

As amizades desenvolvidas durante o curso, em especial aos colegas Gabriel Gutierrez e Rafael Duarte, com os quais tive grande prazer de trabalhar em diversos projetos e atividades.

Ao Prof. Dr. Carlos Danilo Miranda Regis, primeiramente pela amizade, pela orientação, pelas dúvidas geradas e esclarecidas e por possibilitar enorme crescimento intelectual e pessoal durante a pesquisa.

À Profa. Dra. Aganeide Castilho Palitot, que aceitou fazer parte deste trabalho como membro da banca, e muito acrescentou sobre a patologia em estudo.

Ao Prof. Dr. Leonardo Vidal Batista, que aceitou fazer parte deste trabalho como membro da banca, e muito acrescentou no conhecimento sobre aprendizagem de máquinas.

Ao Prof. Dr. Waslon Terllizzie Araújo Lopes, que aceitou fazer parte deste trabalho como membro da banca, e muito acrescentou nos conhecimentos sobre processamento de sinais e por diversos apontamentos e correções.

*"A ciência, meu jovem, é feita de erros,
mas erros que devem ser cometidos,
pois pouco a pouco conduzem à verdade."*

Jules Verne

RESUMO

A retinopatia diabética é uma das principais causas de cegueira e sua detecção é uma tarefa complexa que demanda tempo e experiência do profissional, podendo divergir o prognóstico a depender dos examinadores que efetuem a avaliação. Neste trabalho, objetivando auxiliar na graduação e marcação da patologia, é proposto um *script* em Python baseado em técnicas de redes neurais convolucionais (CNN), com preocupação em preservar as características geométricas das imagens, avaliando os resultados de forma objetiva e subjetiva. A tarefa de graduação se trata de uma avaliação estritamente objetiva e se dá por meio das métricas de acurácia, área sob a curva, revocação e precisão. Na análise subjetiva, é realizada comparação da saída gráfica do código com imagens disponíveis em base de dados marcadas por quatro diferentes especialistas. Uma análise objetiva também é proposta para a avaliação das marcações por meio da acurácia e de nova métrica aqui proposta. Para estas tarefas, são exploradas diferentes arquiteturas de CNN, como Densenet121, Resnet50 e Xception, buscando realizar a extração de características das imagens provenientes da fundoscopia. A etapa de classificação é realizada por meio da inserção das características vindas da CNN em uma rede densa de múltiplas camadas cognitivas (MLP), que irá classificar a imagem em um dos cinco graus da doença, incluindo sua ausência. Técnicas de busca aleatória e otimização da etapa MLP do modelo foram empregadas com intuito de melhorar os resultados, bem como a rede foi treinada com um ajuste fino, diminuindo as camadas ativas e alterando a taxa de aprendizagem. Suportando a generalização do modelo de aprendizagem de máquina, foram realizadas operações de aumento de dados, onde foram geradas imagens com espelhamento horizontal, alteração da posição, proximidade da imagem, brilho e também da cor. Como resultados, foi obtida uma área sob a curva, no melhor caso, de aproximadamente 91%, enquanto que a grafia da área patológica se assemelha a marcações realizadas por médicos oftalmologistas com uma acurácia dos *pixels* superior a 83% para a imagem analisada, também no melhor caso.

Palavras-chave: Retinopatia diabética. Processamento de imagens. Classificação automática. CNN. Redimensionamento. Marcação da área patológica.

ABSTRACT

Diabetic retinopathy is one of the main causes of blindness and its detection is a complex task that demands time and experience from the professional, being possible to have different prognosis depending on the examiners who carry out the evaluation. In this work, with the objective of helping in the grading and marking of the pathology, a Python script based on convolutional neural networks (CNN) techniques is proposed, with a concern to preserve the geometric characteristics of the images, and evaluating the results objectively and subjectively. The grading task is a strictly objective evaluation and is carried out using the metrics of accuracy, area under the curve (AUC), recall and precision. In the subjective analysis, the graphical output of the code is compared with images available in a database with spelling performed by four different specialists. An objective analysis is also proposed for the evaluation of markings through accuracy and new proposed metric. For these tasks, different CNN architectures are explored, such as Densenet121, Resnet50 and Xception, seeking to extract features from fundoscopies images. The classification step is performed by inserting the features coming from the CNN into a multilayer perceptron (MLP), which will classify the image into one of the five degrees of the disease, including its absence. Random search techniques and optimization of the model's MLP step were employed in order to improve the results, as well as the network was trained with a fine adjustment, decreasing the active layers and changing the learning rate. Supporting the generalization of the machine learning model, data augmentation operations were performed, where images were generated with horizontal mirroring, changing position, image proximity, brightness and also color. As a result, an AUC was obtained, in the best case, of approximately 91%, while the pathological area is similar to markings made by ophthalmologists with an accuracy of *pixels* greater than 83% for the image analysed, also in the best case.

Keywords: Diabetic retinopathy. Image processing. Automatic sorting. CNN. Resizing. Pathological area marking.

LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo dos graus da retinopatia diabética.	23
Figura 2 – Exemplo de microaneurismas em exame de fundo do olho.	24
Figura 3 – Diferentes tipos de hemorragias em exame de fundo do olho.	25
Figura 4 – Diferentes tipos de exudatos em exame de fundo do olho.	26
Figura 5 – Exemplo de perolização venosa em exame de fundo do olho.	26
Figura 6 – Formação de um laço venoso em exame de fundo de olho.	27
Figura 7 – Abordagens de classificação de imagens de retinografia.	28
Figura 8 – Diagrama em blocos de um neurônio artificial.	30
Figura 9 – Diagrama em blocos de rede neural supervisionada.	30
Figura 10 – Diagrama da operação de convolução bidimensional.	33
Figura 11 – Diagrama em blocos do elemento da arquitetura Inception.	34
Figura 12 – Diagrama em blocos da arquitetura VGG16.	36
Figura 13 – Exemplo de curva ROC - $AUC = 0,78$	39
Figura 14 – Comparação entre subajustado, ajustado apropriadamente e sobreajustado.	41
Figura 15 – Relação entre a capacidade e o erro.	42
Figura 16 – Diagrama da solução proposta.	44
Figura 17 – Diagrama em blocos da solução proposta para redimensionamento.	47
Figura 18 – Descrição do problema de redimensionamento de imagens de fundoscopia.	48
Figura 19 – Imagens do conjunto de dados com diferenças exploradas no aumento de dados.	51
Figura 20 – Exemplos dos aumentos de dados.	52
Figura 21 – Diagrama da etapa MLP utilizada na rede neural.	53
Figura 22 – Diagrama em blocos da solução proposta para marcação da área representativa da patologia.	55
Figura 23 – Diagrama de caixas com a distribuição dos resultados das métricas para a rede Densenet121.	58
Figura 24 – Diagrama de caixas com a distribuição dos resultados das métricas para a rede Resnet50.	58
Figura 25 – Diagrama de caixas com a distribuição dos resultados das métricas para a rede Xception.	59
Figura 26 – Curva ROC dos resultados obtidos.	60
Figura 27 – Matriz de confusão para a etapa de teste da classificação patológica ou não.	60
Figura 28 – Saída do algoritmo.	61
Figura 29 – Imagem grafada por profissional.	62

Figura 30 – Sobreposição das marcações patológicas.	63
Figura 31 – Sobreposição da máscara do algoritmo para diferentes quantidades de marcações profissionais.	64

LISTA DE TABELAS

Tabela 1 – Estado da arte sobre a detecção da retinopatia diabética.	19
Tabela 2 – Distribuição das diferentes classes na base de dados DRDC2015. .	45
Tabela 3 – Distribuição das diferentes resoluções na base de dados DRDC2015.	45
Tabela 4 – Informação espacial das imagens da Figura 18.	49
Tabela 5 – Relação largura/altura para as diferentes resoluções.	49
Tabela 6 – Descrição das imagens geradas no aumento de dados.	51
Tabela 7 – Descrição dos hiper-parâmetros para busca aleatória.	54
Tabela 8 – Resultados das métricas de validação para as diferentes arquiteturas.	56
Tabela 9 – Resultado da etapa de teste da classificação dos graus da patologia.	59
Tabela 10 – Sumários das métricas da marcação.	65

SUMÁRIO

1	INTRODUÇÃO	15
1.1	Objetivo geral	16
1.1.1	Objetivos específicos	16
1.2	Estado da arte	17
1.3	Estrutura do texto	19
2	RETINOPATIA DIABÉTICA	21
2.1	Diagnóstico suportado por computadores	26
3	APRENDIZAGEM DE MÁQUINA	29
3.1	Arquiteturas de redes convolucionais	32
3.1.1	Redimensionamento	35
3.2	Métodos de interpretação de aprendizagem de máquina	37
3.3	Métricas objetivas	38
3.4	Otimização	40
4	METODOLOGIA	43
4.1	Base de dados	43
4.2	Redimensionamento das Imagens	46
4.3	Aumento de dados	49
4.4	Rede neural	52
4.5	Métodos de interpretação do modelo	54
5	RESULTADOS EXPERIMENTAIS	56
5.1	Classificação da patologia	56
5.2	Análise da Máscara Patológica Proposta	61
5.2.1	Análise subjetiva	61
5.2.2	Análise objetiva	63
6	CONCLUSÃO	66
6.1	Contribuições	67
6.2	Sugestões para trabalhos futuros	67
	REFERÊNCIAS	69
	APÊNDICE A – CÓDIGO PARA AUMENTO DE DADOS	73

APÊNDICE B – CÓDIGO PARA MARCAÇÃO DA ÁREA PATOLÓGICA	76
APÊNDICE C – CÓDIGO PARA TREINAMENTO DA REDE NEURAL	79
APÊNDICE D – SUMÁRIO DA REDE NEURAL TESTADA	83
APÊNDICE E – CÓDIGO PARA MEDIÇÃO DAS MÉTRICAS FINAIS	84
APÊNDICE F – ARTIGO PUBLICADO	85
ANEXO A – CÓDIGO PARA REDIMENSIONAMENTO DAS IMAGENS (RIBEIRO; REGIS, 2022)	86

1. INTRODUÇÃO

A diabetes, ou diabetes mellitus, é uma das emergências globais com crescimento acelerado do Século 21 (QUMMAR *et al.*, 2019; FEDERATION, 2021), sendo uma doença que atinge cerca de 537 milhões de pessoas no mundo (FEDERATION, 2021). Esta é uma condição crônica e séria, que ocorre quando os níveis de glicose se elevam no sangue em razão de uma incapacidade do corpo de produzir insulina em quantidade suficiente, ou empregá-la de forma eficiente.

Esta condição ocorre em dois diferentes tipos: a diabetes tipo 1, ocasionada por um processo autoimune que leva ao ataque das chamadas células Beta no pâncreas, que são responsáveis pela produção da insulina, decorrendo em pouca ou nenhuma produção do hormônio (FEDERATION, 2021); e a diabetes tipo 2, que tem maior prevalência, corresponde a cerca de 90% de todos os casos diagnosticados ao redor do mundo (FEDERATION, 2021), e ocorre, inicialmente, como uma incapacidade do corpo de responder à insulina de maneira esperada, tal qual uma resistência ao hormônio (FEDERATION, 2021).

A retinopatia diabética (RD), conhecida em inglês como *Diabetic retinopathy*, é uma das mais comuns patologias que atingem a retina (WAN; LIANG; ZHANG, 2018) e encontra-se entre as mais frequentes causas de cegueira (CONGDON; FRIEDMAN; LIETMAN, 2003; WAN; LIANG; ZHANG, 2018), atingindo cerca de 4,8% (OTTAIANO *et al.*, 2019). Por se tratar de uma condição progressiva comum a pacientes portadores de diabetes, sendo esperado que cerca de um terço destes sejam também diagnosticados com essa patologia ocular (LAM *et al.*, 2018; QUMMAR *et al.*, 2019), a literatura sugere que pacientes com diabetes realizem acompanhamento periódico da retinopatia duas vezes ao ano (WAN; LIANG; ZHANG, 2018).

O método usual de diagnóstico da retinopatia diabética é feito por meio do exame de imagens coloridas do fundo do olho (WAN; LIANG; ZHANG, 2018), chamado na literatura e pelos oftalmologistas de fundoscopia. Neste procedimento o especialista atenta à presença de microaneurismas, hemorragias e exudatos (duros e moles). De acordo com a existência e intensidade dessas características, o profissional gradua a patologia em cinco diferentes graus. A saber:

1. Sem retinopatia diabética;
2. Retinopatia diabética leve não proliferativa;
3. Retinopatia diabética moderada não proliferativa;
4. Retinopatia diabética severa não proliferativa;

5. Retinopatia diabética proliferativa.

Outros métodos são apontados na literatura como capazes de contribuir para a detecção da RD, sendo alguns destes direcionados à análise dos vasos, como a angiografia fluorescente (AF - *fluorescein angiography*) onde é aplicado um contraste que requer uma análise cuidadosa do médico para evitar problemas posteriores; e a angiografia por tomografia de coerência óptica (*optical coherence tomography angiography* - OCTA) (EL-BAZ; SURI, 2020a), que é uma técnica com custos superiores à fundoscopia. Sendo o último empregado na detecção de estágios iniciais da RD e glaucoma (EL-BAZ; SURI, 2020a).

A detecção da RD em seus estágios iniciais tem um papel crítico para um bom prognóstico (EL-BAZ; SURI, 2020b; LAM *et al.*, 2018; PIRES *et al.*, 2019), no entanto, sua graduação é uma tarefa que demanda um longo tempo para ser realizada (LAM *et al.*, 2018; WAN; LIANG; ZHANG, 2018). Esta complexidade pode ocasionar erros durante a avaliação (WAN; LIANG; ZHANG, 2018), gerando discrepâncias entre resultados (LAM *et al.*, 2018), a depender do nível de habilidade de leitura do profissional.

Dado que há um elevado número de pessoas com diagnóstico de diabetes, a RD pode não ser avaliada e tratada em tempo hábil devido à dependência de processos manuais (GOH *et al.*, 2016; WAN; LIANG; ZHANG, 2018). Desta forma, é reduzida a possibilidade de atuação em tempo de evitar sequelas graves, como a perda da visão, decorrentes de um processo degenerativo da patologia, que pode ser controlado ou retardado quando identificado em seus estágios iniciais (WAN; LIANG; ZHANG, 2018).

1.1 OBJETIVO GERAL

O objetivo deste trabalho consiste em desenvolver uma ferramenta para suporte a detecção e graduação da retinopatia diabética a partir da implementação de algoritmos de processamento de imagens e redes neurais, permitindo também efetuar a marcação de áreas relevantes à análise do profissional oftalmologista no processo de diagnóstico.

1.1.1 Objetivos específicos

- Estudar os aspectos teóricos referentes à retinopatia diabética;
- Estudar e implementar os algoritmos de processamento digital de imagens;
- Estudar e implementar os algoritmos de redes neurais convolucionais;

- Aplicar técnicas de otimização para as redes neurais;
- Avaliar o desempenho dos algoritmos implementados por meio de métricas objetivas;
- Efetuar a identificação de segmentos da imagem que sejam representativos para a detecção da patologia;
- Avaliar as segmentações realizadas pelo algoritmo com marcações realizadas por médicos oftalmologistas.

1.2 ESTADO DA ARTE

Nesta seção, serão apresentados outros trabalhos que foram realizados para a detecção da retinopatia diabética em imagens de fundo do olho, que empregam técnicas de processamento digital de imagens, assim como trabalhos que empregam aprendizagem de características (*feature learning*) e aprendizado profundo, bem como otimização deste tipo de rede neural.

Para esta revisão foram considerados relevantes apenas os trabalhos publicados no últimos cinco anos (a partir de 2018). Para classificar os artigos mais relevantes dentro do vasto universo de publicações, foram considerados prioritárias a quantidade de citações, que demonstram sua relevância dentro do campo em análise, a utilização da mesma base de dados de imagens utilizada neste trabalho. Os artigos selecionados também abordam diferentes premissas para o tratamento da imagem antecedendo sua inserção na rede neural.

O estudo trazido por Lam *et al.* (2018) suportou-se nas bases de dados Messidor-1 e MildDR, para efetuar uma classificação que identificasse a presença ou não da RD. A rede neural empregada na tarefa utilizava a arquitetura GoogLeNet. Para contornar as problemáticas da aquisição de dados ser pouco uniforme e do tamanho da base disponível para o treinamento, foram empregadas técnicas de aumento de dados (*data augmentation*), realizando as operações de zoom, rotação aleatória e translação aleatória. Foi ainda empregada a técnica de pré-processamento com melhoramento do histograma chamada de *contrast limited adaptive histogram equalization (CLAHE)*. O resultado obtido por estes autores foi de uma acurácia de 74,5%.

Também objetivando avaliar a presença ou ausência da patologia, Pires *et al.* (2019) utilizaram os conjuntos de dados: das imagens do desafio do ano de 2015 de Detecção da Retinopatia Diabética do Kaggle (*Diabetic Retinopathy Detection Challenge 2015 - DRDC2015*), DR2 e Messidor-2. Eles fizeram uso de uma rede similar à desenvolvida pelo time "o_O", durante o DRDC2015, que era baseada na arquitetura VGG, com uso de três resoluções: 128×128 , 256×256 e 512×512 .

Foram empregadas técnicas de aumento de dados com zoom, espelhamento, rotação aleatória, translação aleatória, alongamento e deslocamento de valores vermelho, verde, azul (*red, green, blue* - RGB). O desempenho foi avaliado no *dataset* DRDC2015 onde a área sob a curva (*Area Under the Curve* - *AUC*) obtida foi de 95,5%.

Ainda buscando verificar de forma binária o problema, Tymchenko, Marchenko e Spodarets (2020) utilizaram os conjuntos de dados: DRDC2015, IDRiD, Messidor e APTOS. A rede neural arquitetada pelos autores foi um *ensemble* com as arquiteturas EfficientNet-B4 e B5 e SE-ResNeXt50 usando técnicas de classificação, regressão linear e regressão ordinal. Para aumentar a quantidade de dados disponível foram empregadas técnicas de aumento de dados como: distorção óptica, distorção de grade, transformação afim por partes, espelhamento, inversão vertical, rotação aleatória, deslocamento aleatório, escala aleatória, um deslocamento de valores RGB, brilho e contraste aleatórios, ruído gaussiano aditivo, desfoque, nitidez, relevo, gama aleatória e cortar fora. A métrica de desempenho representativa para este trabalho foi a acurácia onde foi obtido o valor de 99,3%.

Na pesquisa de Wan, Liang e Zhang (2018) foram utilizadas as imagens do DRDC2015 (TYMCHENKO; MARCHENKO; SPODARETS, 2020), para graduar a ausência ou presença da RD em cinco graus. Para realizar a tarefa foi empregada uma rede VggNet-s, em conjunto, também, com uma série de operações de aumento de dados, como: espelhamento, rotação aleatória e alongamento. Uma etapa de melhoramento das imagens foi realizada em um pré-processamento com a operação de filtragem de médias não locais (*NonLocal Means Denoising* - *NLMD*), e por fim o trabalho atingiu uma acurácia de 95,7%.

Qummar *et al.* (2019) também utilizou o *dataset* DRDC2015, também para uma classificação multiclases. A rede neural empregada por estes autores empregou um *ensemble* com as arquiteturas de redes convolucionais Resnet50, Inveception_v3, Xception, Densenet121 e Densenet169. O aumento de dados foi realizado por meio das operações de: espelhamento, rotação aleatória e normalização. Foi feito um *Up-sampling* com recortes de 512×512 *pixels* e *down-sampling* com descarte de imagens para igualar a quantidade de imagens de cada classe a quantidade disponível para a classe menos presente, resultando, por fim, numa acurácia de 80,8%.

Ainda buscando classificar o problema nas suas diversas graduações Gangwar e Ravi (2021) utilizaram duas bases de dados: APTOS 2019 e Messidor-1. A arquitetura de rede neural empregada foi Inception-ResNet-v2. O método empregado no trabalho faz uma filtragem da imagem, busca as coordenadas da área de delimitação e faz o recorte de uma porção quadrada da imagem. A avaliação do modelo foi realizada nas duas bases de dados obtendo acurácia de 82,18% no *dataset* APTOS 2019 e 72,33%

no Messidor-1.

Por fim, Zia *et al.* (2022) utilizou da base de dados DRDC2015, para a mesma tarefa de classificação. A arquitetura da rede neural proposta no trabalho utilizava uma combinação da VGG e da Inception_v3 usando técnicas de seleção de dados, redimensionamento, extração de características, fusão de características e seleção de características. A métrica de desempenho utilizada para avaliar o trabalho foi a acurácia, atingindo o valor de 96,4%, também foi analisado no trabalho o tempo de processamento.

O resumo dos trabalhos utilizados como estado da arte para esta dissertação pode ser visto na Tabela 1.

Tabela 1 – Estado da arte sobre a detecção da retinopatia diabética.

Autor	Base de dados	Quantidade de classes	Métrica de avaliação
Lam <i>et al.</i> (2018)	Messidor-1 e MildDR	Binário	Acurácia de 74,5%
Wan, Liang e Zhang (2018)	DRDC2015	Cinco classes	Acurácia de 95,7%
Pires <i>et al.</i> (2019)	DRDC2015, DR2, Messidor-2	Binário	AUC 95,5% no <i>dataset</i> DRDC2015
Qummar <i>et al.</i> (2019)	DRDC2015	Cinco classes	Acurácia de 80,8%
Tymchenko, Marchenko e Spodarets (2020)	DRDC2015, IDRiD, Messidor e APTOS	Binária	Acurácia de 99,3%
Gangwar e Ravi (2021)	APTOS e Messidor-1	Cinco classes	Acurácia de 82,2% no <i>dataset</i> APTOS
Zia <i>et al.</i> (2022)	DRDC2015	Cinco classes	Acurácia de 96,4%

1.3 ESTRUTURA DO TEXTO

Esta dissertação de mestrado está organizada em seis capítulos que compreendem a teoria associada ao objeto de estudo da pesquisa, assim como a metodologia empregada e os resultados iniciais do trabalho.

No primeiro capítulo é realizada uma introdução sobre a retinopatia diabética, bem como da pesquisa em geral, apontando os objetivos do trabalho e um breve relato sobre o estado da arte.

No Capítulo 2 a patologia em estudo é apresentada em detalhes, discorrendo sobre as causas da retinopatia diabética, como ela é graduada e seus principais elementos característicos. O Capítulo 3 aborda uma introdução ao aprendizado de máquinas, técnicas de regularização, arquiteturas de redes neurais convolucionais e otimização são discutidas de forma breve uma vez que são importantes etapas da metodologia deste trabalho.

O Capítulo 4 aborda a descrição da metodologia empregada. É feita uma comparação entre o uso ou não da técnica de redimensionamento com preservação das formas.

O Capítulo 5 contém os resultados obtidos para as simulações de algumas arquiteturas de redes neurais. Em seguida, no último capítulo estes resultados serão analisados e discutidos, findando em propostas de próximos passos a serem seguidos na linha desta pesquisa.

2. RETINOPATIA DIABÉTICA

A retinopatia diabética é uma doença ocular crônica que pode levar a perda irreversível da visão (EL-BAZ; SURI, 2020b; CONGDON; FRIEDMAN; LIETMAN, 2003) devido a complicações nos microvasos (PIRES *et al.*, 2019). Contudo, a fundoscopia, método mais usado para sua detecção, requer a atuação de especialistas e trata-se de um processo longo e complexo.

A diabetes é uma desordem na metabolização de açúcar que atinge uma em cada 11 pessoas no mundo (PIRES *et al.*, 2019). A literatura aponta que cerca de um terço da população diagnosticada com esta doença desenvolve retinopatia diabética (RD - *diabetic retinopathy*) (LAM *et al.*, 2018), havendo ainda evidências apontando que indivíduos com excesso de glicose no sangue estão 25 vezes mais propensos a perder a visão por consequência de complicações da RD (PIRES *et al.*, 2019).

Alguns fatores podem levar ao desenvolvimento da RD, notavelmente a longevidade da diabetes, um controle glicêmico pouco efetivo, bem como a hipertensão (BANDELLO *et al.*, 2018). Sendo o controle glicêmico apontado como a melhor forma de reduzir a sua incidência e evolução (BANDELLO *et al.*, 2018).

O diagnóstico da retinopatia diabética é obtido por meio da análise do fundo do olho, uma técnica que permite a análise da retina. Contudo, esta depende de um vasto conjunto de características, havendo necessidade de avaliar inclusive a localização em que os sinais surgem (LAM *et al.*, 2018). A Organização Mundial de Saúde recomenda que esse exame seja realizado pelo menos uma vez ao ano (PIRES *et al.*, 2019), havendo ainda recomendação para duas vezes ao ano (WAN; LIANG; ZHANG, 2018) para pessoas acometidas pela diabetes.

Discernir a presença de microaneurismas, pequenos desprendimentos saculares nos capilares (*saccular outpouching of capillaries*), hemorragias e rupturas dos vasos na fundoscopia é uma tarefa árdua, em especial nos graus iniciais, e autores apontam requerer atenção e expertise do profissional avaliador (EL-BAZ; SURI, 2020b; LAM *et al.*, 2018).

As imagens de fundoscopia são aplicáveis à detecção de diabetes, glaucoma, neoplasia do olho (EL-BAZ; SURI, 2020b), algumas vantagens são relacionadas na literatura para este método, sendo destacáveis (EL-BAZ; SURI, 2020b):

- Aquisição diretamente no consultório;
- Não necessita aplicação de contraste;
- Visualização real da retina;

- Observa um vasto campo da retina em uma única imagem em comparação a oftalmoscopia indireta;
- Permite o acompanhamento da doença;
- Baixo custo em comparação a outras técnicas.

Por outro lado, algumas limitações são relatadas para as retinografias. Por se tratar de uma imagem bidimensional, não é possível avaliar a profundidade das estruturas como nos exames de tomografia, assim, a avaliação das estruturas é limitada, as imagens apresentam menos zoom e clareza, condições como catarata podem dificultar a interpretação da imagem, além de que o processo de aquisição pode produzir imagens incomuns (EL-BAZ; SURI, 2020b).

A retinopatia diabética ocorre em diferentes estágios. Uma classificação usual a divide em quatro, dos quais três são chamados de não proliferativos (*nonproliferative diabetic retinopathy* - NPDR) e um proliferativo (*proliferative diabetic retinopathy* - PDR), que é o grau mais avançado da patologia. O termo proliferativo é oriundo da neovascularização que ocorre nesse estágio, levando a vazamento de soro, hemorragia e proliferação fibrovascular na interface vítrea da retina, o que resulta ainda em hemorragia vítrea e descolamento de retina por tração (OSAADON *et al.*, 2014). Em seu livro, El-Baz e Suri (2020b) adiciona ainda a categoria edema macular.

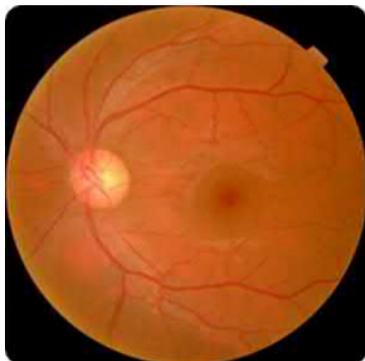
A partir da Figura 1 é possível observar nas sub-figuras 1a e 1b, imagens obtidas de pacientes sem diagnóstico da patologia. Na Figura 1c observa-se o estágio leve não proliferativo, a Figura 1d é um exemplo de caso moderado não proliferativo, enquanto a Figura 1e demonstra um caso severo não proliferativo. Por fim, a Figura 1f representa um caso proliferativo.

Para a detecção da patologia nos seus graus não proliferativos é realizada a busca por alterações microvasculares e mudanças no interior da retina durante o exame de fundoscopia (BANDELLO *et al.*, 2018). Essas mudanças são caracterizadas pela presença de microaneurismas, hemorragias, exudatos, exudatos algodonosos (*cotton wool spots*), perolização venosa, edema macular e anormalidades nos microvasos (EL-BAZ; SURI, 2020b; BANDELLO *et al.*, 2018).

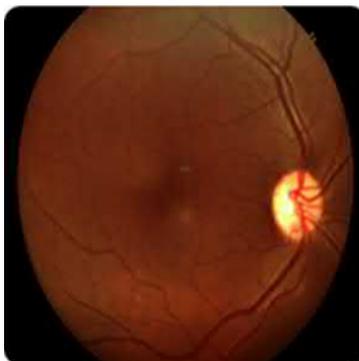
Com a evolução da patologia, tendem a surgir sintomas como visão borrada, pontos cegos, visão central distorcida, perda de visão (EL-BAZ; SURI, 2020b). Estas manifestações reforçam a importância da detecção ainda nos estágios iniciais, com um diagnóstico assertivo, incluindo sua graduação como forma de minimizar as complicações e a possibilidade de perda de visão (EL-BAZ; SURI, 2020b; LAM *et al.*, 2018; PIRES *et al.*, 2019)

Figura 1 – Exemplo dos graus da retinopatia diabética.

(a) paciente 1 sem diagnóstico da patologia.



(b) paciente 2 sem diagnóstico da patologia.



(c) estágio leve não proliferativo.



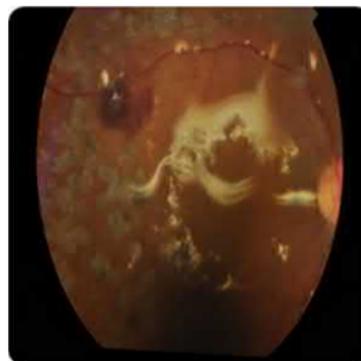
(d) estágio moderado não proliferativo.



(e) caso severo não proliferativo.



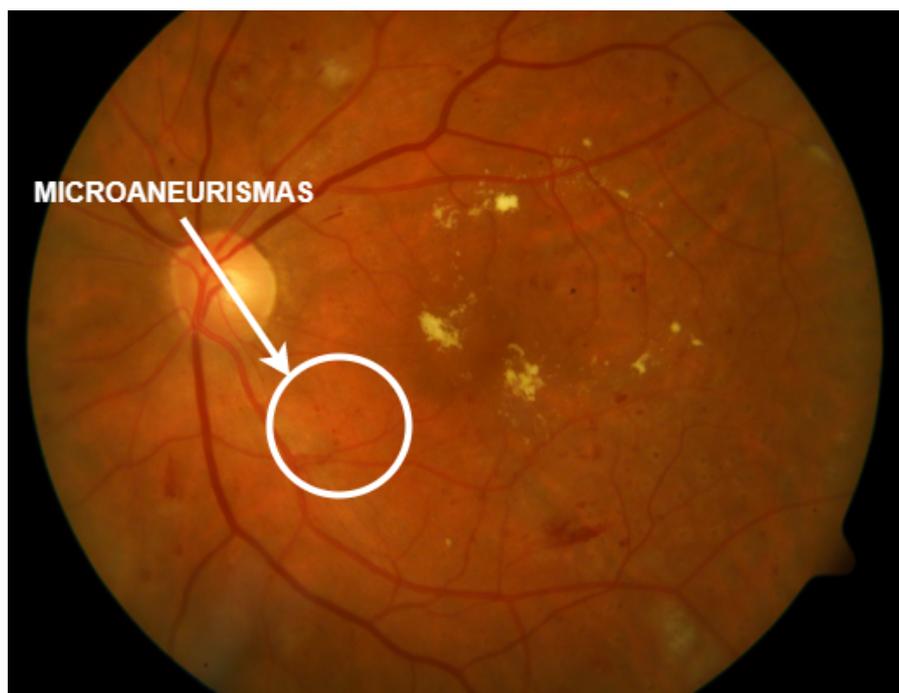
(f) estágio proliferativo.



Fonte: El-Baz e Suri (2020b, p. 141).

Os microaneurismas são apontados como um dos primeiros estágios da doença, e se caracterizam pela formação de estruturas em forma de balões, externas as paredes dos capilares, e pela dilatação das arteríolas ou pequenas vênulas (BANDELLO *et al.*, 2018). Eles costumam ser assintomáticos (EL-BAZ; SURI, 2020b; BANDELLO *et al.*, 2018), e podem desaparecer com o tempo. Na Figura 2 vê-se uma seta apontando para um microaneurisma. Esta imagem mostra a dificuldade no reconhecimento dos elementos relevantes. Exames de maior complexidade, como tomografia de coerência óptica (*optical coherence tomography* - OCT) e angiotomografia de coerência óptica estão sendo empregados atualmente para analisar a morfologia dessas estruturas (BANDELLO *et al.*, 2018).

Na pesquisa de Bandello *et al.* (2018) é trazida a informação de que técnicas computadorizadas para o cálculo do aparecimento e desaparecimento dos micro-

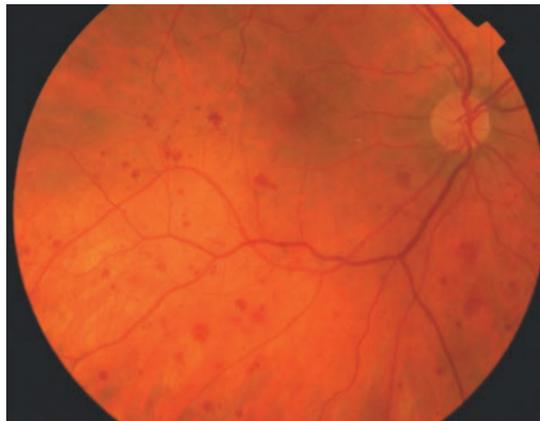
Figura 2 – Exemplo de microaneurismas em exame de fundo do olho.

Adaptado de: DiaretDB1(KÄLVIÄINEN; UUSITALO, 2007).

aneurismas vem sendo empregadas para a avaliação do estágio da RD, uma vez que uma taxa elevada indica um maior risco de progressão da patologia. Esta taxa apresenta ainda uma correlação com o aumento da espessura da mácula. Estes mesmos autores descrevem que as hemorragias podem ser causadas pela ruptura dos microaneurismas, vazamento de capilares e anormalidades dos microvasos.

Ainda em seu texto, Bandello *et al.* (2018) descrevem que uma imagem de fundoscopia, estas estruturas aparecem com formato de chama ou como pontos (manchas), a depender das regiões em que elas são identificadas. Os autores caracterizam que as hemorragias com formato de chama ocorrem paralelas à superfície da retina, enquanto que os pontos são presentes no sentido perpendicular a ela. As hemorragias em formato de mancha são mais profundas sendo apontado como um dos sinais da isquemia retinal (SCANLON *et al.*, 2009). Na Figura 3a destaca-se a presença de uma hemorragia com padrão de chama, enquanto na Figura 3b é destacado o padrão de pontos e manchas.

Outras estruturas presentes na imagem do fundo do olho são os exudatos duros. Estes são estruturas amareladas que se formam em conjunto e frequentemente circunvizinham microaneurismas (BANDELLO *et al.*, 2018). A detecção dessas estruturas em proximidade à borda de uma área de espessamento da retina ocasiona perda de acuidade visual (BANDELLO *et al.*, 2018), tornando assim essa uma condição sintomática da RD. Um exemplo desta estrutura pode ser visto na Figura 4a, evidenciando-se

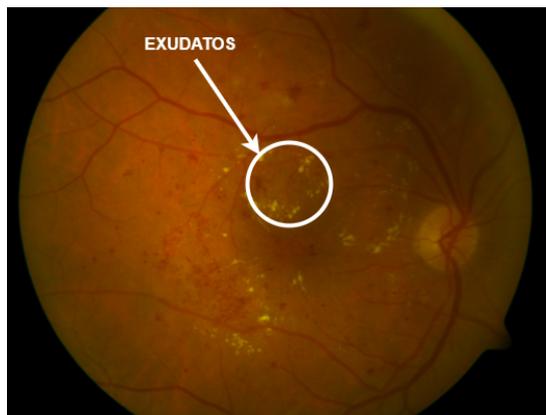
Figura 3 – Diferentes tipos de hemorragias em exame de fundo do olho.**(a) hemorragia em formato de chama.****(b) hemorragia em formato de ponto e mancha.**Fonte: Scanlon *et al.* (2009, p. 34).Adaptado de: Bandello *et al.* (2018, p. 29).

no tom amarelado que difere dos microaneurismas e as hemorragias.

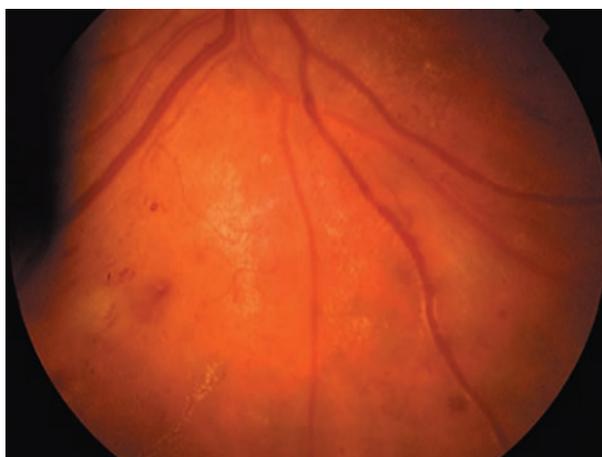
Enquanto os exudatos duros são caracterizados por serem estruturas amareladas com aspecto de cera e se apresentam como coleções eosinofílicas de fluido proteico (CORRÊA; JR, 2005), tem-se também os exudatos algodonosos (*cotton wool spots*)¹ que são similares a essas estruturas, porém tem coloração acinzentada (BANDELLO *et al.*, 2018). Corrêa e Jr (2005) caracterizam os exudatos moles pelo edema focal da camada de fibras nervosas originário de células ganglionares infladas por axoplasma estagnado. Alguns autores se referem a estes como exudatos moles (JAYA; DHEEBA; SINGH, 2015), sendo comum o seu aparecimento nas proximidades de vasos sanguíneos (BANDELLO *et al.*, 2018). Um exemplo de imagem de fundoscopia contendo *cotton wool spots* pode ser visto na Figura 4b.

Alterações da espessura dos vasos, como visto na Figura 5, caracterizam a perolização venosa (*venous beading*) (BANDELLO *et al.*, 2018). Esse tipo de alteração vascular é comum de ser observada em regiões de oclusão de capilares, sendo uma característica de estágios avançados da retinopatia diabética não proliferativa (BANDELLO *et al.*, 2018). Quando um vaso desvia do seu percurso natural, o que pode ser percebido por meio da análise histórica do paciente, ele dá voltas, ou forma laços, apresentando derivações abruptas de seus caminhos esperados - um problema característico da patologia em estudo. Um exemplo disto pode ser visto na Figura 6.

¹ São manchas que lembram chumaços de algodão, também são apresentados na literatura como sendo exudatos moles.

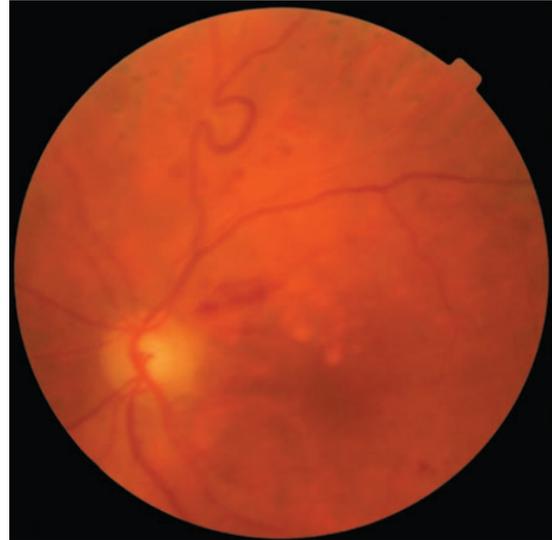
Figura 4 – Diferentes tipos de exudatos em exame de fundo do olho.**(a) exudatos duros.****(b) exudatos algodonosos.****Adaptado de: DiaretDB1 (KÄLVIÄINEN; UUSITALO, 2007).**

A Figura 6a apresenta o olho de um paciente antes do acontecimento da derivação abrupta, enquanto na Figura 6b está presente uma estrutura similar a letra grega ômega maiúscula na parte superior da imagem. Anormalidades genéricas presentes nos microvasos também são possíveis em pacientes com RD.

Figura 5 – Exemplo de perolização venosa em exame de fundo do olho.**Fonte: Scanlon *et al.* (2009, p. 38).**

2.1 DIAGNÓSTICO SUPORTADO POR COMPUTADORES

O diagnóstico suportado por computadores da RD é explorado como maneira de reduzir a sobrecarga nos oftalmologistas e diminuir a inconsistência na avaliação entre especialistas (MOOKIAH *et al.*, 2013). Para o suporte à detecção ser realizado de forma a obter um resultado assertivo, diversas técnicas como agrupamento com

Figura 6 – Formação de um laço venoso em exame de fundo de olho.**(a) imagem anterior ao desenvolvimento.****(b) imagem com o laço presente.****Fonte: Scanlon *et al.* (2009, p. 36).**

K-Médias, modelos de campos aleatórios de Markov, filtragem Kalman no trabalho de Rossant *et al.* (2009) e diferentes abordagens de segmentação (ROSSANT *et al.*, 2009; YAZDANPANAHI *et al.*, 2010; CHEN *et al.*, 2016; KAFIEH *et al.*, 2013; EHNES *et al.*, 2014) foram empregadas como etapa anterior à tarefa de classificação.

Técnicas de detecção automática, quando aplicadas em especial no início da doença, oferecem uma oportunidade única para a prevenção da perda da visão ocasionada pela retinopatia diabética, bem como o uso de técnicas de aprendizagem de máquina com redes neurais convolucionais também permitem a detecção de sinais indicativos de outros tipos de patologias oculares (LAM *et al.*, 2018).

O trabalho de Lam *et al.* (2018) corrobora que as imagens das classes moderada e severa da RD possuem elementos de maior tamanho nas imagens, o que torna mais fácil a identificação destas categorias pelas redes neurais convolucionais (*Convolutional Neural Network - CNN*), uma vez que a transferência de aprendizado foi otimizada para classificação de conteúdos macroscópicos.

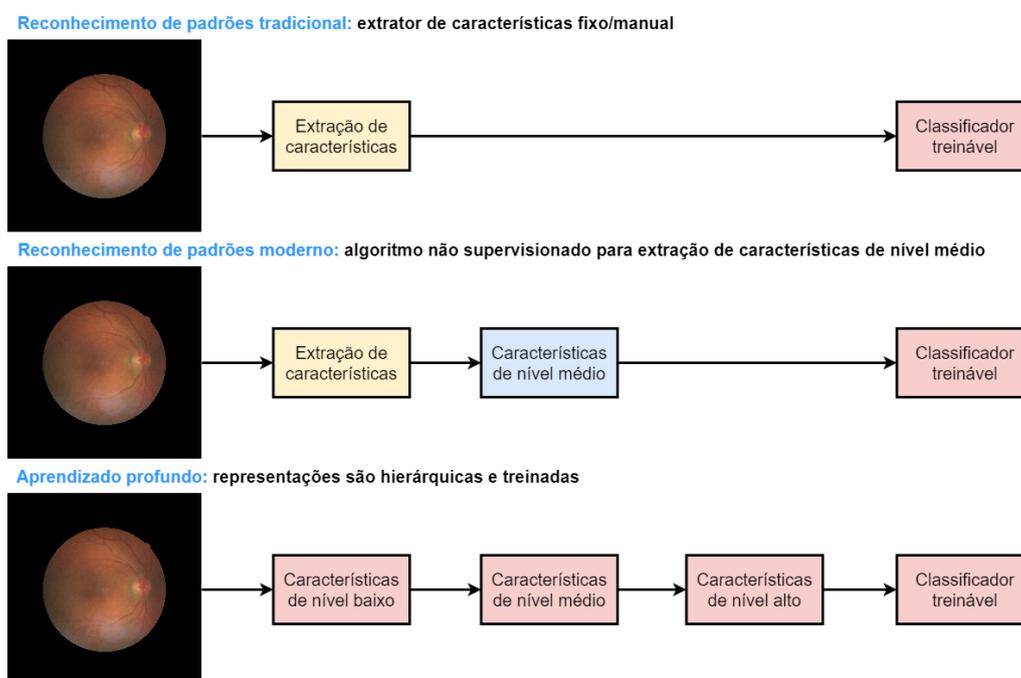
É oportuno que se destaque que características sutis podem ser cruciais no diagnóstico de patologias em imagens médicas, como é o caso em imagens de grau leve da RD. Durante os trabalhos de (LAM *et al.*, 2018) obteve-se sensibilidade de 95% durante a fase de validação.

As técnicas tradicionais em reconhecimento de padrões de imagens utilizavam uma etapa de extração de características com algoritmos específicos, que antecederiam

a tarefa de classificação (EL-BAZ; SURI, 2020a). Os algoritmos modernos envolvem a etapa de extração pensada ao processo de transformação destas *features* em uma forma otimizada, com características de nível intermediário (EL-BAZ; SURI, 2020a).

Por último e mais recentemente, a etapa de extração de características foi removida, sendo executada pela própria rede neural dentro nas camadas convolucionais (EL-BAZ; SURI, 2020a). Nos algoritmos de aprendizagem profunda é possível obter características de baixo, médio e alto nível para treinar o classificador. A diferença entre os modelos foi diagramada por El-Baz e Suri (2020a), como pode ser visto na Figura 7.

Figura 7 – Abordagens de classificação de imagens de retinografia.



Adaptado de: El-Baz e Suri (2020a, p. 28).

Percebe-se desta forma que a correta identificação da retinopatia diabética nos estágios iniciais é de suma importância para a qualidade de vida dos pacientes acometidos pela diabetes, uma vez que cerca de 33% dos portadores de diabetes apresentam esta comorbidade (LAM *et al.*, 2018; QUMMAR *et al.*, 2019).

3. APRENDIZAGEM DE MÁQUINA

Aprendizagem de máquina ou *machine learning* é o termo genérico utilizado para as técnicas que permitem o aprendizado da máquina a partir dos dados, considerando-se a definição de aprendizado como a capacidade de converter experiência em expertise ou conhecimento (GOODFELLOW; BENGIO; COURVILLE, 2016). Estas técnicas que utilizam o acúmulo de experiências, em geral, são empregadas para solucionar problemas complexos, como as tarefas realizadas por animais ou seres humanos (SHALEV-SHWARTZ; BEN-DAVID, 2014), ou mesmo que excedam as capacidades humanas, como o processamento de grandes volumes de dados. Há ainda a aplicação em problemas em que existe a necessidade de adaptabilidade (SHALEV-SHWARTZ; BEN-DAVID, 2014).

Pode-se diferenciar a programação direta e o aprendizado de máquina pela capacidade de ser flexível ao longo do tempo, ou seja, de ser adaptável. Essa característica de desenvolver a plasticidade do aprendizado, de forma similar à existente no cérebro humano, é a premissa base das chamadas redes neurais artificiais, ou simplesmente redes neurais. Enquanto a programação *hand-on*, permite a solução de um problema como ele se apresenta, as redes neurais permitem o aprendizado com base nos estímulos a que ela é submetida (HAYKIN, 2008).

O elemento base desenvolvido para tentar replicar a capacidade de aprendizado do cérebro humano é o neurônio artificial. Este é definido em Haykin (2008) como a unidade de processamento de informação fundamental. A Figura 8 retrata o diagrama em blocos dessa estrutura, onde cada uma das entradas, que são chamadas sinapses, é ponderada e somada a uma polarização, *bias*, que uma vez quantificada, é passada para uma função de ativação que vai ser responsável por gerar um estímulo de saída.

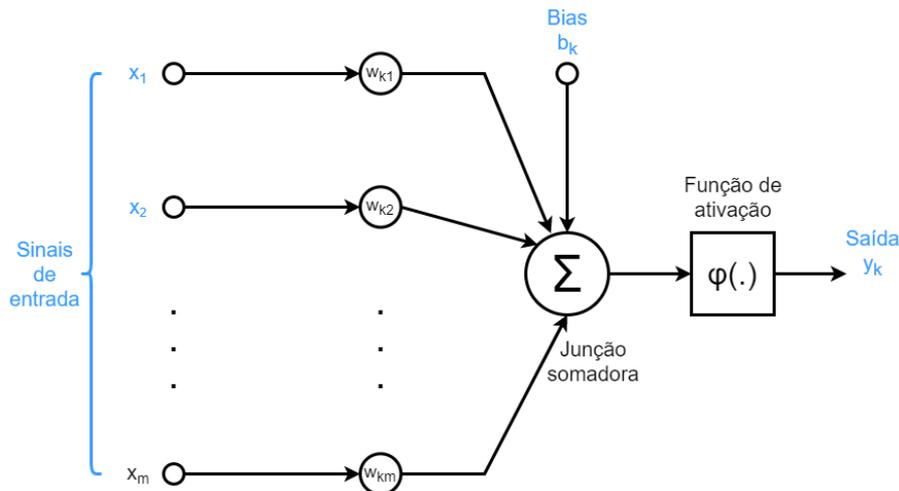
A saída do neurônio k exposto na Figura 8, na n -ésima iteração, pode ser quantificada a partir da Equação 1. De forma simplista podemos descrever matematicamente que o peso sináptico da entrada j , do neurônio artificial k , w_{kj} , sofre um ajuste Δw_{kj} , definido como na Equação 2 (HAYKIN, 2008).

$$y_k(n) = \varphi\left(\sum_{j=1}^m (x_j(n) \cdot w_{kj}(n)) + b_k\right) \quad (1)$$

$$\Delta w_{kj}(n) = \eta \cdot y_k(n) \cdot x_j(n) \quad (2)$$

Aqui, o n representa a iteração do algoritmo, η é referente à taxa de aprendizagem, que pode ser constante ou variar ao longo do treinamento, φ é a função de

Figura 8 – Diagrama em blocos de um neurônio artificial.

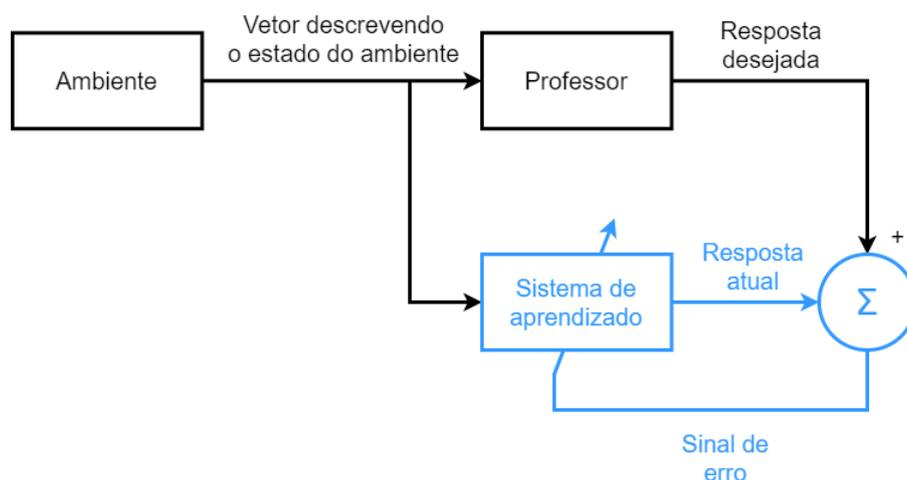


Adaptada de: Haykin (2008, p. 11).

ativação do neurônio, $x_j(n)$ e $y_k(n)$ são os sinais de excitação e pós-sinápticos na n -ésima iteração, respectivamente.

De forma geral, os algoritmos são divididos com base em dois paradigmas, supervisionado ou não supervisionado (HAYKIN, 1999). Durante o processo de aprendizagem supervisionado, o algoritmo é exposto a um conjunto de entradas e saídas (HAYKIN, 1999), de maneira que os parâmetros da rede são ajustados baseados na combinação de influências dos estímulos e respostas obtidos (HAYKIN, 1999). Dessa forma, o aprendizado supervisionado pode ser visto como um processo fechado com realimentação (HAYKIN, 2008), conforme visto na Figura 9.

Figura 9 – Diagrama em blocos de rede neural supervisionada.



Adaptada de: Haykin (2008, p. 35).

Por outro lado, no aprendizado não supervisionado, também conhecido como auto-organizado, as informações sobre as saídas ótimas, ou desejadas, não são

repassadas de forma explícita para a rede. Implica a inexistência de rótulos sobre a função a ser aprendida por ela (HAYKIN, 2008). Os parâmetros da rede são ajustados com base nas estatísticas dos dados de entrada (HAYKIN, 2008), levando a um fluxo direto dos dados.

Algoritmos de aprendizagem de máquina são empregados em diferentes tarefas, como: classificação, regressão, transcrição, tradução automática, saída estruturada, detecção de anormalidade, síntese e amostragem, imputação de valores perdidos, remoção de ruído, estimativa de densidade ou estimativa de função de massa de probabilidade (GOODFELLOW; BENGIO; COURVILLE, 2016).

O desafio central na aprendizagem de máquina é atingir um preditor generalista, que obtenha desempenho satisfatório com base em dados novos e nunca vistos, e não apenas nos quais o modelo foi treinado (GOODFELLOW; BENGIO; COURVILLE, 2016). Ainda, é apresentado que esta área vem demonstrando uma das evoluções mais rápidas dentro da ciência da computação (SHALEV-SHWARTZ; BEN-DAVID, 2014).

Há situações em aplicações de aprendizagem de máquina em que a quantidade de características à disposição para treinar o modelo é expressiva, sendo interessante reduzir esse conjunto para usar no preditor. Este problema é conhecido por *feature selection* (SHALEV-SHWARTZ; BEN-DAVID, 2014). Os preditores que utilizam apenas um pequeno conjunto de características requerem menos memória e podem ser aplicados de forma mais rápida (SHALEV-SHWARTZ; BEN-DAVID, 2014).

Ainda, em alguns casos tem-se um dado bruto, como uma imagem, e precisa-se extrair características. Este processo de automação da extração de características é conhecido por *feature learning*.

Uma vez obtidas as *features* a serem utilizadas pelo preditor, elas podem ser usadas de forma a melhorar o desempenho deste. A regularização é, então, o processo pelo qual alguma alteração é realizada com o objetivo de reduzir o erro de generalização, porém não o erro de treinamento (GOODFELLOW; BENGIO; COURVILLE, 2016; SHALEV-SHWARTZ; BEN-DAVID, 2014). Várias técnicas de regularização ou normalização estão disponíveis, como: centralização, recorte (*clipping*), limitação unitária (*unit range*), normalização da média (*standardization*) (SHALEV-SHWARTZ; BEN-DAVID, 2014).

É ainda importante destacar que técnicas conhecidas como aprendizado profundo já foram empregadas no diagnóstico, detecção e graduação de patologias como Alzheimer, nódulos torácicos, doenças pulmonares (EL-BAZ; SURI, 2020b), COVID-19.

3.1 ARQUITETURAS DE REDES CONVOLUCIONAIS

As redes neurais convolucionais (*convolutional neural network* - CNN/ConvNet) se apresentaram como uma solução robusta dentro da visão computacional nos últimos anos (CHOLLET, 2017). Estas redes iniciaram como empilhamento simples de operações de convolução para extração de características junto a operações de *max-pooling* para uma subamostragem espacial (CHOLLET, 2017). Por volta de 2014, a qualidade das arquiteturas teve uma melhoria significativa por meio do uso de redes mais profundas e extensas (SZEGEDY *et al.*, 2016).

Esta forma de rede se trata de um dos tipos mais populares de rede neural de aprendizado profundo (*deep neural network*) (ALBAWI; MOHAMMED; AL-ZAWI, 2017). Uma das mais importantes suposições, senão a mais importante, é a de que este tipo de rede é capaz de extrair características independente da localização onde elas se apresentem nas imagens (ALBAWI; MOHAMMED; AL-ZAWI, 2017). Essa característica é particularmente relevante na análise da fundoscopia, uma vez que os indícios da patologia, como os microaneurismas e os exudatos, podem ocorrer em diferentes posições. As CNN apresentam a vantagem de serem capazes de processar grandes quantidades de dados e desta forma encontram aplicações em classificação de imagens e processamento de linguagem natural (*natural language processing* - NLP) (JOSHI *et al.*, 2021). Algumas arquiteturas já são bem conhecidas na literatura, dentre as quais destacam-se a Densenet121 (QUMMAR *et al.*, 2019), a Resnet50 (WAN; LIANG; ZHANG, 2018; QUMMAR *et al.*, 2019) e a Xception (QUMMAR *et al.*, 2019), que foram as topologias empregadas na tarefa de extração de características das imagens de fundoscopia deste trabalho.

Segundo Szegedy *et al.* (2016), as CNNs que obtêm bons resultados na classificação de imagens na ImageNet também apresentam vantagens em outras aplicações de visão computacional que requerem recursos visuais de alta qualidade. A ImageNet é uma base de dados que contém milhões de imagens anotadas e que tem sido usada como referência para avaliar o desempenho de sistemas de classificação em larga escala (SIMONYAN; ZISSERMAN, 2014).

As *ConvNets* se tornaram então uma comodidade no campo da visão computacional (SIMONYAN; ZISSERMAN, 2014). A evolução destas redes se deu não só na acurácia de classificação e localização dentro do ImageNet, como também as tornou aplicáveis a outros conjuntos de dados, nos quais atingiram uma excelente performance, mesmo quando usadas como parte de uma *pipeline* relativamente simples, como por exemplo a classificação de características profundas por meio de uma máquina de vetor de suporte (SIMONYAN; ZISSERMAN, 2014).

Como dito anteriormente, o aprendizado de características nas CNN pode ser do tipo supervisionado ou não supervisionado. Usa-se o não supervisionado quando os pesos da arquitetura não podem ser alterados. Desta maneira a rede se encarrega de aprender características de baixo e alto nível em imagens médicas (EL-BAZ; SURI, 2020b). Contudo, quando os pesos das camadas convolucionais podem ser ajustados durante um treinamento, usa-se um processo do tipo supervisionado, de forma a tentar aprimorar a capacidade da rede de extrair *features* relevantes ao problema em estudo.

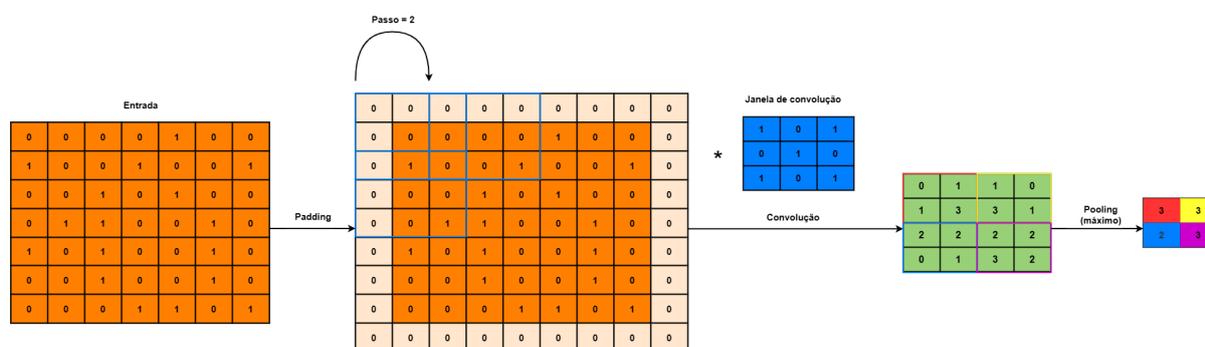
Nessas arquiteturas as redes são construídas com diversas camadas, como convolucionais, *pooling*, e neurônios conectados. A saída de cada uma dessas camadas é submetida a uma função de ativação ou a um mapeamento de forma a ser apresentada como entrada da camada posterior (EL-BAZ; SURI, 2020b).

A camada convolucional realiza a operação de convolução bidimensional, da imagem x , de tamanho $H \times W$, para cada uma dos três canais RGB, d , utilizando um núcleo, f , originando uma matriz y .

O *pooling* usa uma notação derivada da camada convolucional, em que o elemento x^l é a entrada da primeira etapa da camada de *pooling*. Neste momento não existe aprendizado, logo, seu peso é nulo (EL-BAZ; SURI, 2020a).

A operação de *pooling* vai mapear cada subparte em um único valor (EL-BAZ; SURI, 2020a). O *max pooling* então representa o maior valor da subseção em análise.

Figura 10 – Diagrama da operação de convolução bidimensional.



Adaptada de: Li *et al.* (2021, p. 2).

Na Figura 10 encontra-se um diagrama explicativo das operações de uma rede convolucional. O primeiro passo em uma rede como essas é realizar o ajuste na matriz de entrada por meio de um preenchimento (*padding*). Subsequente a isso, a janela de convolução vai ser aplicada sobre a matriz original de forma a obter os coeficientes da nova matriz, que no desenho é representada em verde.

O deslocamento entre as operações de convolução sobre a matriz com os complementos pode ser variável, representando um passo (*stride*). Na sequência, a

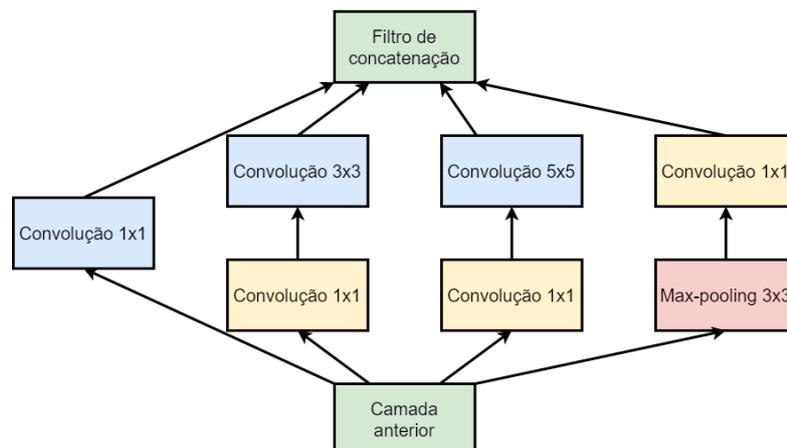
convolução é submetida à operação de *pooling*. No caso representado na imagem, a janela da operação é 2×2 do tipo máximo, esse passo gera uma subamostragem da matriz em verde.

As CNN servem como algoritmos de *feature learning* para imagens, e com base na indicação de Shalev-Shwartz e Ben-David (2014), é possível supor que diferentes arquiteturas podem prover melhores características e diminuir o esforço do classificador.

As CNN já foram empregadas na classificação da retinopatia diabética (ELBAZ; SURI, 2020b). Diversas arquiteturas de redes neurais convolucionais foram propostas ao longo dos anos, destacam-se a Inception (SZEGEDY *et al.*, 2016), Xception (CHOLLET, 2017), Densenet (HUANG *et al.*, 2017), Resnet (HE *et al.*, 2016) e VGG (SIMONYAN; ZISSERMAN, 2014).

A Inception foi uma das arquiteturas precursoras, servindo inclusive de base para a Xception. A hipótese fundamental para o desenvolvimento desta arquitetura é que as correlações entre canais e as espaciais são suficientemente desacopladas, de forma que é preferível não mapeá-las conjuntamente (CHOLLET, 2017). Assim, o módulo básico da Inception é voltado para fazer o processo mais simples e eficiente fatorando a imagem explicitamente em uma série de operações que examinem independentemente estes tipos de correlações (CHOLLET, 2017). O elemento básico da arquitetura Inception é representado na Figura 11 (SZEGEDY *et al.*, 2015). Este módulo é muito similar a uma convolução separável em profundidade (*depthwise separable convolution*), que pode ser vista como uma convolução espacial realizada de forma independente em cada um dos canais (CHOLLET, 2017).

Figura 11 – Diagrama em blocos do elemento da arquitetura Inception.



Adaptada de: Szegedy *et al.* (2015, p. 4).

De forma a evoluir a ideia da Inception, a Xception analisa a possibilidade das correlações entre canais e espaciais serem não apenas suficientemente, mas profun-

damente desacopladas, dessa forma a hipótese é que o mapeamento da correlação entre canais e as correlações espaciais nos mapas de características de uma rede neural convolucional podem ser completamente apartadas (CHOLLET, 2017). Essa arquitetura então é inteiramente baseada em convolução separável em profundidade (CHOLLET, 2017). Chollet (2017) define de forma breve que a Xception é uma pilha linear de camadas de convoluções separáveis em profundidade com conexões residuais. O autor expõe que esta rede é relevante aos estudos pois durante testes com a base de dados ImageNet, apresentou desempenho de classificação superior aos da VGG16 e da InceptionV3.

Outra arquitetura relevante aos estudos é a VGG. Essa estrutura investiga os efeitos na acurácia de redes convolucionais profundas (SIMONYAN; ZISSERMAN, 2014), a partir do uso de filtros de convolução pequenos 3×3 . A parte convolucional dessa rede com 16 camadas de pesos (VGG16) foi sintetizada na Figura 12, e é exposto que esta arquitetura apresenta resultados de acurácia aproximados ao obtido com a InceptionV3 e Xception (CHOLLET, 2017). Uma variação prática dessa rede é a VGG19, que possui três camadas de pesos a mais que a VGG16. A VGG é uma arquitetura que quando submetida a avaliação no momento da sua proposição, fixa o tamanho da imagem em 224×224 pixels em RGB (SIMONYAN; ZISSERMAN, 2014), e essa resolução se torna importante por ser utilizada como padrão em diversos ensaios, inclusive no presente trabalho.

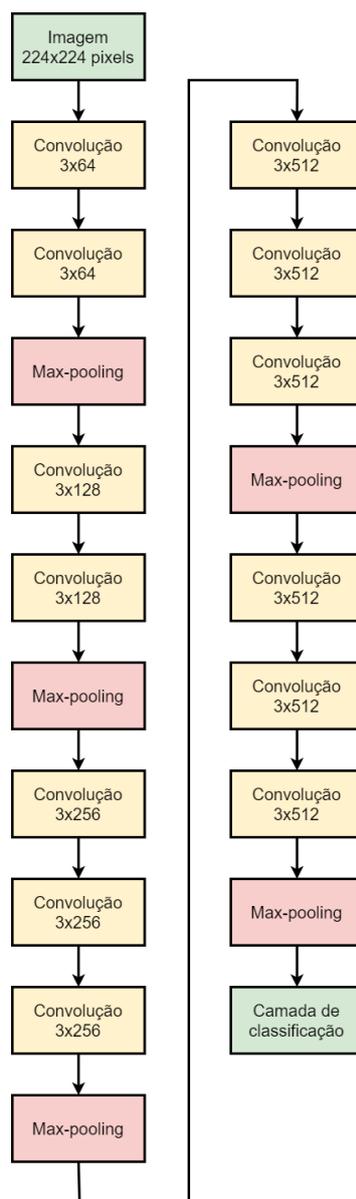
Uma vez que as arquiteturas de redes neurais são extensas, treinar os pesos dessas redes se torna uma tarefa demorada, e por isso diversas aplicações se beneficiam da transferência de pesos do conjunto ImageNet. O conjunto em questão compreende milhares de imagens representando mamíferos, pássaros, peixes, répteis, anfíbios, veículos, mobília, instrumentos musicais, formações geológicas, ferramentas, flores e frutas (DENG *et al.*, 2009). Uma das premissas envolvidas no desenvolvimento da base de dados é que os pesos aprendidos para a classificação de objetos comuns possam servir para o aprendizado de características de objetos raros (DENG *et al.*, 2009).

3.1.1 Redimensionamento

LeCun, Bengio *et al.* (1995) em seu trabalho ressaltaram que nas situações em que a extração de características acontecem com o uso de redes convolucionais em substituição aos métodos manuais, ainda assim se mantinham os requisitos de tamanho e orientação, mesmo que aproximados, das imagens.

As características de compartilhamento de pesos e subamostragem típicas das redes neurais permitem um teor de invariância dos resultados em respeito a pequenas

Figura 12 – Diagrama em blocos da arquitetura VGG16.



Baseado na Tabela 1 de: Simonyan e Zisserman (2014, p. 3).

variações na geometria ou distorções (LECUN; BENGIO *et al.*, 1995).

A literatura aponta para o recorte das imagens do *dataset* em tamanho quadrado, contudo, esta técnica pode acarretar na remoção de informações relevantes posicionadas nos contornos do globo ocular, além disso não é relatado se uma mesma imagem passa por múltiplos recortes.

A questão de uma mesma imagem originar múltiplas imagens é preocupante, uma vez que pode ocasionar vazamento de imagens do conjunto de treinamento para o conjunto de teste ou da validação. Inclusive, como forma de garantir que não existam interações entre os conjuntos, a literatura aplica as técnicas de aumento de dados

posteriormente à divisão da base de dados.

Uma medida de avaliação da imagem com base na capacidade humana de diferenciação dos tons de cinza é chamada de informação espacial (*Spatial Information* - SI) (GONZALEZ; WOODS, 2008). O cálculo desta operação é apresentado em (REGIS, 2013) onde F é uma imagem em escala de cinza, G_y é o gradiente vertical, que contém as bordas no sentido vertical, na direção y , G_x é o gradiente horizontal, que contém as bordas no sentido vertical, na direção x e std é a operação de desvio padrão.

$$Sobel(F) = \sqrt{G_x^2 + G_y^2} \quad (3)$$

$$SI(F) = std[Sobel(F)] \quad (4)$$

3.2 MÉTODOS DE INTERPRETAÇÃO DE APRENDIZAGEM DE MÁQUINA

Os códigos de aprendizagem de máquinas são muitas vezes considerados uma caixa preta, mas métodos de interpretação destes estão sendo assunto de diversas pesquisas atuais (ZHAO *et al.*, 2020). Lundberg e Lee (2017) destacam que é de suma importância a capacidade de interpretar o modelo de predição. Entender o modelo é importante pra confiabilidade, capacidade de ação, responsabilidade, depuração, entre outras tarefas (LUNDBERG; ERION; LEE, 2018).

Para modelos simples, a melhor explicação é ele próprio (LUNDBERG; LEE, 2017), no entanto, técnicas de análise precisam ser aplicadas para sistemas complexos, como o aprendizado profundo ou modelos combinados (LUNDBERG; LEE, 2017). Usando um método de explicação simples, que pode ser entendida como qualquer aproximação interpretável do modelo original (LUNDBERG; LEE, 2017). Zhao *et al.* (2020) ainda advertem que esses métodos de análise não são facilmente aplicados a dados desestruturados.

As técnicas existentes para aplicação em dados estruturados são: diagnóstico baseado em modelo substituto, diagnóstico baseado em valores Shapley e diagnóstico baseado em gradiente (ZHAO *et al.*, 2020). Um exemplo do modelo baseado em substituto é o *Local Interpretable Model-agnostic Explanation (LIME)*, o qual é comumente usado (ZHAO *et al.*, 2020). O modelo baseado em valores Shapley, como o *SHapley Additive exPlanations (SHAP)*, é um método de diagnóstico local. Como representantes dos métodos de gradiente tem-se Gradientes Integrados, DeepLIFT (ZHAO *et al.*, 2020).

No método LIME, dado um ponto de interesse no espaço do preditor, ele leva a uma vizinhança adequada de pontos em torno deste e se adequa a um modelo local

simples que pode ser interpretável (ZHAO *et al.*, 2020).

Três métodos clássicos utilizam os valores Shapley, originalmente voltados para a teoria dos jogos cooperativos e posteriormente empregados na análise de modelos (LUNDBERG; LEE, 2017). O SHAP então foi proposto por Lundberg e Lee (2017) como uma medida unificada da importância de uma característica. No SHAP, cada predição é analisada em contribuições individuais e por esse motivo Lundberg, Erion e Lee (2018) destacam que é um cálculo de alto custo computacional.

Os métodos baseados em gradiente são especialmente rápidos de serem calculados em redes neurais, nas quais os gradientes estão disponíveis. Nesses métodos, um produto entre o gradiente e o valor da característica é um ponto de partida na determinação da importância de cada atributo (ZHAO *et al.*, 2020).

Com o suporte das técnicas de interpretação de predições, a tensão crescente entre a acurácia e a interpretabilidade do modelo preditor pode ser contornada (LUNDBERG; LEE, 2017), sendo motivadora da busca por estratégias mais robustas.

3.3 MÉTRICAS OBJETIVAS

A avaliação objetiva dos resultados obtidos por meio de uma rede neural, é realizada por meio da investigação das diferentes medidas de treinamento desta.

A acurácia é a razão entre o número de acertos e o número de amostras disponíveis, como visto na Equação 5, na qual VP são os verdadeiros positivos, VN são os verdadeiros negativos, FP são os falsos positivos e FN são os falsos negativos. A revocação é matematicamente calculada como na Equação 6; e a precisão descreve-se como na Equação 7.

$$ACC = \frac{VP + VN}{VP + VN + FP + FN} \quad (5)$$

$$Revocação = \frac{VP}{VP + FN} \quad (6)$$

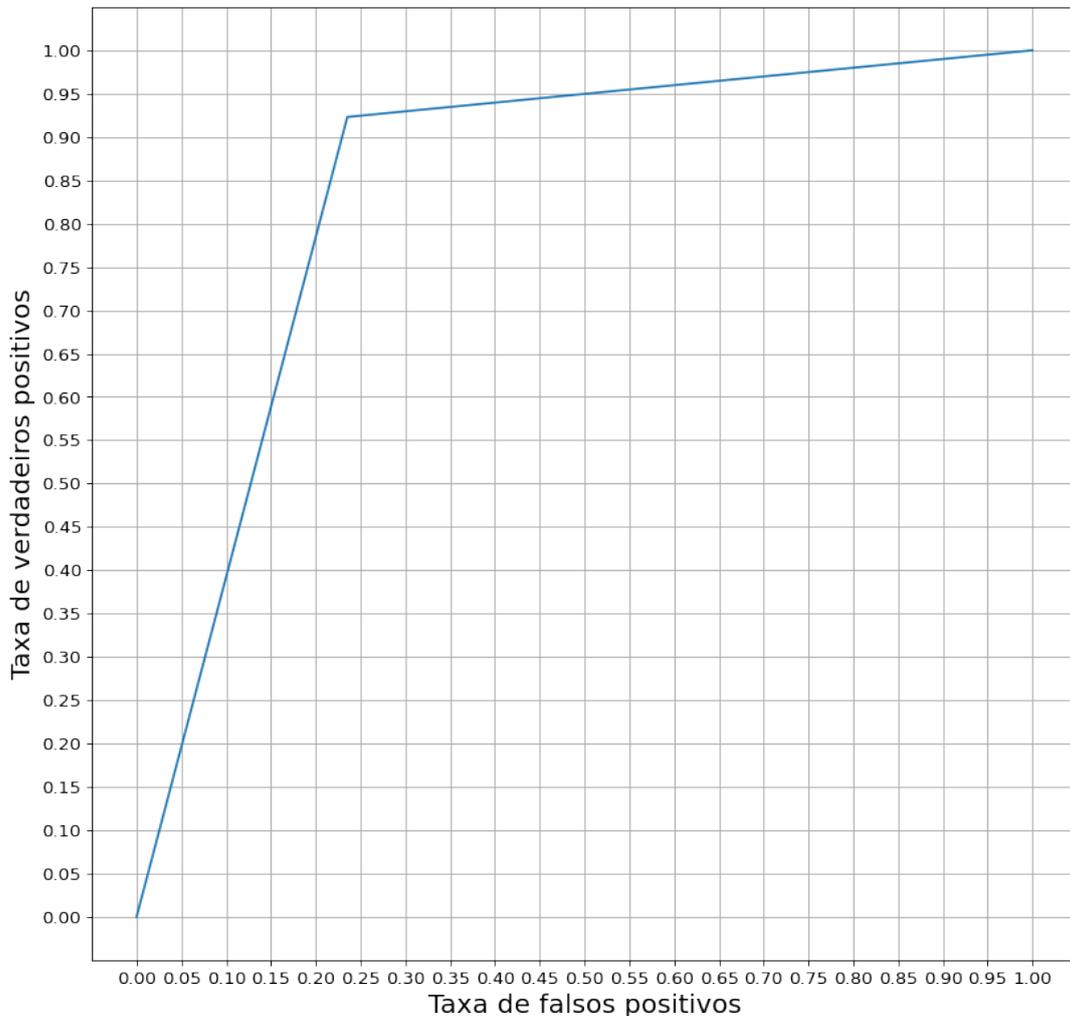
$$Precisão = \frac{VP}{VP + FP} \quad (7)$$

De maneira geral, a revocação e a precisão são medidas que se complementam. O aumento de uma tende a piorar a outra, o que leva à busca de um equilíbrio entre essas medidas.

A área sob a curva (*Area Under the Curve* - AUC) representa o grau de separabilidade das diferentes classes (QUMMAR *et al.*, 2019). A medida de AUC é calculada

com base na curva de característica de operação do receptor (*receiver operating characteristic* - ROC). Na Figura 13, tem-se a taxa dos verdadeiros positivos no eixo Y (ordenadas) e a taxa de falsos positivos no eixo X (abscissas), e o valor da AUC é a integral do gráfico. A estratégia um contra todos consiste em calcular a curva ROC para cada uma das n -classes. Em cada passo uma dada classe é considerada como a positiva enquanto as demais como um montante negativo.

Figura 13 – Exemplo de curva ROC - $AUC = 0,78$.



Rakotomamonjy (2004) exemplifica que quando a AUC é igual a 1, o classificador obteve a acurácia ideal, enquanto um classificador com AUC de 0,5 é um preditor com características de aleatoriedade. Seguindo essa mesma lógica, tem-se que uma AUC de 0 é um preditor que erra sempre.

O coeficiente inserção sobre união (*Intersection over Union* - IoU), ou Índice de Jaccard, é uma medida da sobreposição entre dois conjuntos, no caso de aprendizagem de máquina ele tem sido empregado para avaliar a sobreposição de duas

máscaras. O coeficiente é definido, no contexto de aprendizagem de máquinas, como na Equação 8. Esse coeficiente varia de 0, quando as máscaras são completamente distintas, a 1 quando as máscaras são completamente iguais.

$$IoU = \frac{VP}{VP + FP + FN} \quad (8)$$

3.4 OTIMIZAÇÃO

A maioria dos algoritmos de *machine learning* tem ajustes conhecidos como hiper parâmetros que precisam ser determinados pelo programador, externamente ao algoritmo de aprendizado (GOODFELLOW; BENGIO; COURVILLE, 2016).

Usualmente há um interesse em quão bem o algoritmo consegue se apresentar para dados não vistos durante a fase de treinamento, porque isso é que determina quão bem ele se sai quando aplicado em situações práticas, do mundo real (GOODFELLOW; BENGIO; COURVILLE, 2016). Desta forma, são realizados testes de performance usando um conjunto de teste separado dos dados utilizados para o treinamento do sistema (GOODFELLOW; BENGIO; COURVILLE, 2016).

Selecionar adequadamente a medida de avaliação de desempenho é frequentemente uma tarefa árdua, uma vez que esta precisa corresponder ao comportamento desejado posteriormente (GOODFELLOW; BENGIO; COURVILLE, 2016). Essa medida de desempenho pode ser obtida dentro do conjunto de treinamento, originando o chamado erro de treinamento (GOODFELLOW; BENGIO; COURVILLE, 2016), que pode ser reduzido ainda durante este momento.

No aprendizado da máquina é desejável que o erro de generalização, chamado também de erro de teste, seja o menor possível (GOODFELLOW; BENGIO; COURVILLE, 2016). Esse seria o erro do sistema quando apresentado a dados do mundo real. A obtenção de resultados satisfatórios é viável mediante a realização de suposições acerca da divisão dos conjuntos (GOODFELLOW; BENGIO; COURVILLE, 2016). Esta divisão deve ser feita com base em uma distribuição de probabilidade da base dos dados disponíveis e os exemplos devem ser independentes (GOODFELLOW; BENGIO; COURVILLE, 2016).

O erro de teste esperado é maior ou igual ao valor esperado do erro de treinamento (GOODFELLOW; BENGIO; COURVILLE, 2016). Assim, para que uma rede neural obtenha um desempenho satisfatório, deve-se reduzir o erro de treinamento e manter o espaçamento entre este e o erro de teste, o menor possível (GOODFELLOW; BENGIO; COURVILLE, 2016). O distanciamento entre as curvas dos erros serve como forma de analisar dois dos desafios existentes durante o treinamento do algoritmo: o

subajustado (*underfitting*) e o sobreajustado (*overfitting*), chamado também de sobre treinamento (*overtraining*) (HAYKIN, 2008).

Underfitting é o termo utilizado para descrever um modelo que não aprendeu o suficiente das amostras de treinamento, apresentando um baixo desempenho no conjunto de treinamento (GOODFELLOW; BENGIO; COURVILLE, 2016). Enquanto que o *overfitting* ocorre quando o espaçamento entre os gráficos de erro do treino e do teste é muito grande.

Goodfellow, Bengio e Courville (2016) apontam que a solução para o *overfit* e *underfit* é alterar a capacidade do modelo, que é a habilidade deste de se ajustar a uma variedade de funções. Os autores trazem que quando um modelo tem baixa capacidade, ele pode ficar preso a um baixo desempenho no conjunto de treinamento, enquanto, modelos com alta capacidade podem memorizar propriedades do conjunto de treinamento que são pouco relevantes dentro do conjunto de teste.

Figura 14 – Comparação entre subajustado, ajustado apropriadamente e sobreajustado.

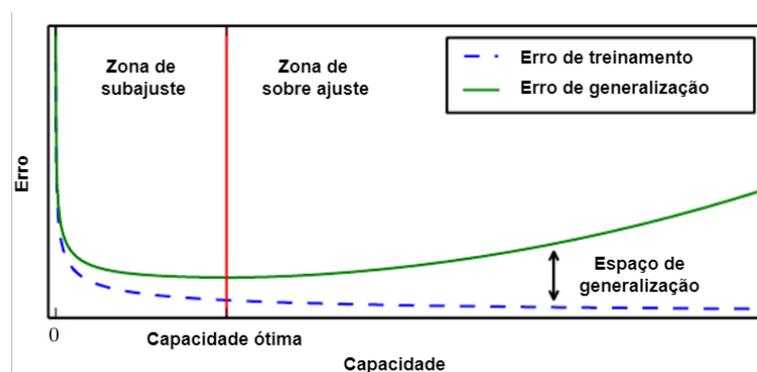


Adaptado de: Shalev-Shwartz e Ben-David (2014, p. 69).

Na Figura 14 vê-se o resultado de algoritmos de aprendizagem de máquina que ficaram subajustado, ajustado apropriadamente e sobre ajustado, na tarefa de representar uma função de terceiro grau. Pode ser visto que no caso sobre ajustado, os pontos passam próximos da curva, mas a modelagem generalista não consegue retratar a concavidade da primeira metade dos pontos. Em oposição, no sobreajustado, a curva se força a passar por todos os pontos, descrevendo uma curva que não representa de forma adequada a função que originou os pontos.

Na Figura 15 observa-se os efeitos da variação da capacidade de um algoritmo e dos erros obtidos na fase de treinamento e de teste, além de ser destacado o espaço de generalização (*generalization gap*). Como dito anteriormente, o melhor desempenho do algoritmo se dá quando existe um pequeno distanciamento entre as curvas, obtendo neste ponto a capacidade ótima do algoritmo. Pode-se visualizar que antes dele há um *underfitting* dos dados, e após, existe o *overfitting*. Uma característica importante de se destacar é que o erro de treinamento evolui muito pouco com o

Figura 15 – Relação entre a capacidade e o erro.



Adaptado de: Goodfellow, Bengio e Courville (2016, p. 115).

aumento da capacidade após uma determinada quantidade de épocas com base no gráfico de evolução do treinamento, levando o código a consumir muitos recursos para uma pequena evolução.

Haykin (2008) discorre sobre duas formas de treinar uma rede perceptron de múltiplas camadas (*multilayer perceptron* - MLP), sendo uma em lote (*batch*) e outra estocástica (sequencial), que também é chamada de *online* (GOODFELLOW; BENGIO; COURVILLE, 2016). Contudo, o termo *online* se destina aos casos em que os exemplos são repassados com base em um fluxo de dados constantemente alimentado (GOODFELLOW; BENGIO; COURVILLE, 2016).

No treinamento em lotes a sensibilidade da rede é ajustada para o conjunto de treinamento antes dos pesos sinápticos da rede (HAYKIN, 2008). Em contraposição, no modo estocástico os pesos sinápticos são ajustados a cada amostra de treinamento (HAYKIN, 2008).

Algumas estratégias para melhorar as métricas que avaliam o aprendizado são (SHALEV-SHWARTZ; BEN-DAVID, 2014): pegar um número maior de amostras, mudar a hipótese para cada classe (seja aumentando, reduzindo, alterando completamente ou mudando os parâmetros a serem considerados), mudar as características de representação dos dados, alterar o algoritmo de otimização utilizado para aplicar a regra de aprendizado.

Segundo Shalev-Shwartz e Ben-David (2014), o erro de aproximação de uma determinada classe não depende da quantidade de amostras, ou do algoritmo utilizado, mas sim da distribuição e da hipótese usada para a classe. Essa observação se faz relevante porque não é possível ampliar o conjunto de treinamento em diversas situações, seja pelo tempo necessário para a aquisição ou por qualquer outro problema. Então a análise do fato que ocasiona a baixa performance deve ser visto à parte do conjunto de dados.

4. METODOLOGIA

Neste capítulo é apresentada a solução proposta para realizar a detecção, gradação e marcação da retinopatia diabética, o método é baseado em técnicas de processamento digital de imagens e aprendizagem de máquina, com emprego de técnicas de extração de características por meio de arquitetura de redes neurais convolucionais. A abordagem principal deste trabalho concentra-se em treinar uma rede com capacidade de efetuar a marcação de áreas relevantes ao profissional oftalmologista no momento da realização do diagnóstico da patologia.

A solução aqui proposta pode ser vista de forma esquemática na Figura 16. Nesta abordagem serão utilizados dois passos, o primeiro para o treinamento da rede em detecção e gradação da patologia e o segundo para identificar a área em que a patologia se faz presente, de forma a conceder ao oftalmologista uma ferramenta de análise assistida.

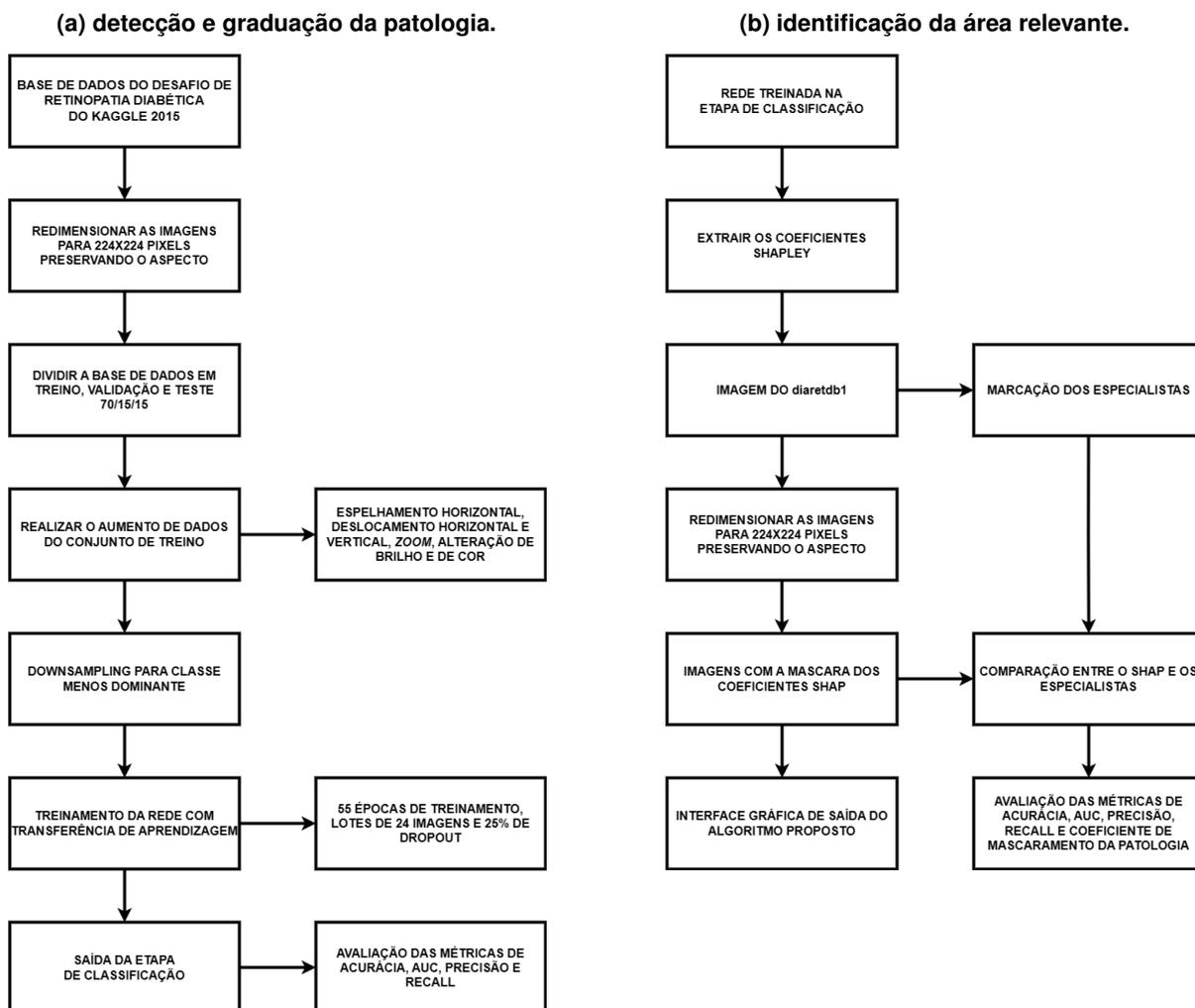
4.1 BASE DE DADOS

Neste trabalho, utilizam-se duas bases de dados. A primeira delas é o conjunto utilizado no desafio do ano de 2015 de Detecção da Retinopatia Diabética do Kaggle (*Diabetic Retinopathy Detection Challenge 2015 - DRDC2015*) (TYMCHENKO; MARCHENKO; SPODARETS, 2020). Este é composto de 35126 imagens de fundos-copia colorida, sendo o maior banco de dados público para a patologia em estudo (TYMCHENKO; MARCHENKO; SPODARETS, 2020). A segunda base de dados é a DiaretDB1, que possui uma marcação realizada por quatro oftalmologistas das áreas representativas de diferentes eventos anômalos da retinopatia diabética no fundo do olho, estas imagens foram capturadas em um hospital universitário de Kuopio (KÄLVIÄINEN; UUSITALO, 2007), num total de 89 arquivos.

O treinamento da rede relacionado ao diagrama da Figura 16a foi realizada por meio código do Apêndice C, enquanto que o diagrama da Figura 16b demonstra em blocos as operações do código do Apêndice B.

No DRDC2015, as imagens são rotuladas em cinco categorias distintas com base na severidade dos sinais da patologia (LAM *et al.*, 2018; QUMMAR *et al.*, 2019), caracterizados como: sem retinopatia diabética (rótulo 0), retinopatia diabética leve (rótulo 1), retinopatia diabética moderada (rótulo 2), retinopatia severa (rótulo 3) e retinopatia proliferativa (rótulo 4) (TYMCHENKO; MARCHENKO; SPODARETS, 2020). As imagens estão divididas nas classes como apresentado na Tabela 2. A base é distribuída de maneira pouco uniforme, tendo a maior representatividade na classe das imagens que não possuem a patologia (mais de 73%), sendo considerado por (QUM-

Figura 16 – Diagrama da solução proposta.



MAR *et al.*, 2019) como perfeitamente desbalanceado. O mesmo autor aponta que o uso de base de dados desbalanceadas em aprendizado profundo leva ao enviesamento do modelo. Em (TYMCHENKO; MARCHENKO; SPODARETS, 2020) é apontado que diferentes bases de dados possuem distribuições similares às base de dados utilizadas neste trabalho, os leva a considerar esta como sendo uma característica intrínseca ao tipo de dados.

As imagens que compõem o banco de dados também são pouco uniformes no que diz respeito ao equipamento utilizado para as aquisições, sendo a resolução de 3888×2952 *pixels* a que possui maior representatividade, ou seja, maior quantidade de imagens disponíveis. A distribuição geral das imagens com base em suas resoluções é apresentada na Tabela 3. Destaca-se que grande parte das imagens possuem uma proporção de tela do tipo panorama, tendo os pesquisadores efetuado

Tabela 2 – Distribuição das diferentes classes na base de dados DRDC2015.

Classe	Rótulo	Quantidade	Percentual da base
não patológica	0	25810	73,48%
leve	1	2443	6,95%
moderada	2	5292	15,07%
severa	3	873	2,49%
proliferativa	4	708	2,02%

Adaptado de Pires *et al.* (2019, p. 99).

o redimensionamento ou recorte destas imagens para atingir um formato quadrado. Algumas das resoluções utilizadas na literatura são 128×128 (PIRES *et al.*, 2019), 256×256 (LAM *et al.*, 2018; PIRES *et al.*, 2019), 380×380 (QUMMAR *et al.*, 2019), 456×456 (TYMCHENKO; MARCHENKO; SPODARETS, 2020) e 512×512 (PIRES *et al.*, 2019; QUMMAR *et al.*, 2019; TYMCHENKO; MARCHENKO; SPODARETS, 2020).

Tabela 3 – Distribuição das diferentes resoluções na base de dados DRDC2015.

Resolução	Quantidade	Percentual da base
3888×2592	10352	29,47%
4752×3168	5550	15,80%
3504×2336	4144	11,80%
2560×1920	3858	10,98%
2592×1944	3795	10,80%
Outras	7427	21,14%

A base de dados então foi particionada em 70% para treinamento, 15% para validação e 15% para teste, com uma divisão estratificada de forma que todas as classes estivessem presentes no mesmo percentual. A separação dos arquivos foi realizada com base no nome do arquivo de imagem, desta forma, o mesmo conjunto de imagens foi apresentado para todos os treinamentos, independentemente da arquitetura empregada.

A base de dados DiaretDB1, contém um total de 89 imagens de fundoscopia coloridas (KÄLVIÄINEN; UUSITALO, 2007). Destas, 84 contém sinais da retinopatia diabética não proliferativa leve e cinco foram consideradas normais pela ausência de sinais, de acordo com a inspeção de todos os especialistas que participaram da etapa de avaliação (KÄLVIÄINEN; UUSITALO, 2007). Nesse banco de dados as imagens foram analisadas por quatro especialistas que efetuaram marcações nos

locais onde eram perceptíveis microaneurismas, hemorragias e exudatos (duros e moles) (KÄLVIÄINEN; UUSITALO, 2007).

Faz-se importante a observação que as marcações presentes na base de dados não são interpretadas da mesma forma por todos os profissionais. Algumas áreas são marcadas de forma mais amplas e outras mais delicadas e detalhadas. As marcações, então, são sobrepostas, gerando áreas com escala de cinza em que a maior intensidade tem cor branca e aparecem nos locais que são grafados por múltiplos profissionais, e áreas menos intensas, representadas de forma mais escuras, nas posições onde apenas um profissional realizou a marcação.

4.2 REDIMENSIONAMENTO DAS IMAGENS

O processo de redimensionamento empregado como pré-processamento pode ser dividido como na Figura 17. A primeira etapa é realizar um ajuste na resolução da imagem (*ori_res*), de forma que se torne quadrada, e a resolução da nova (*new_res*) imagem será múltipla da resolução destino (224×224). Para acrescentar linhas ou colunas na imagem, foi utilizada a Equação 9, definindo quantas linhas e colunas serão acrescentadas em cada direção da imagem. As linhas e colunas criadas são formadas com *pixels* zeros.

$$lin, col = \frac{new_res - ori_res}{2} \quad lin, col \in \mathbb{Z}_+, \quad (9)$$

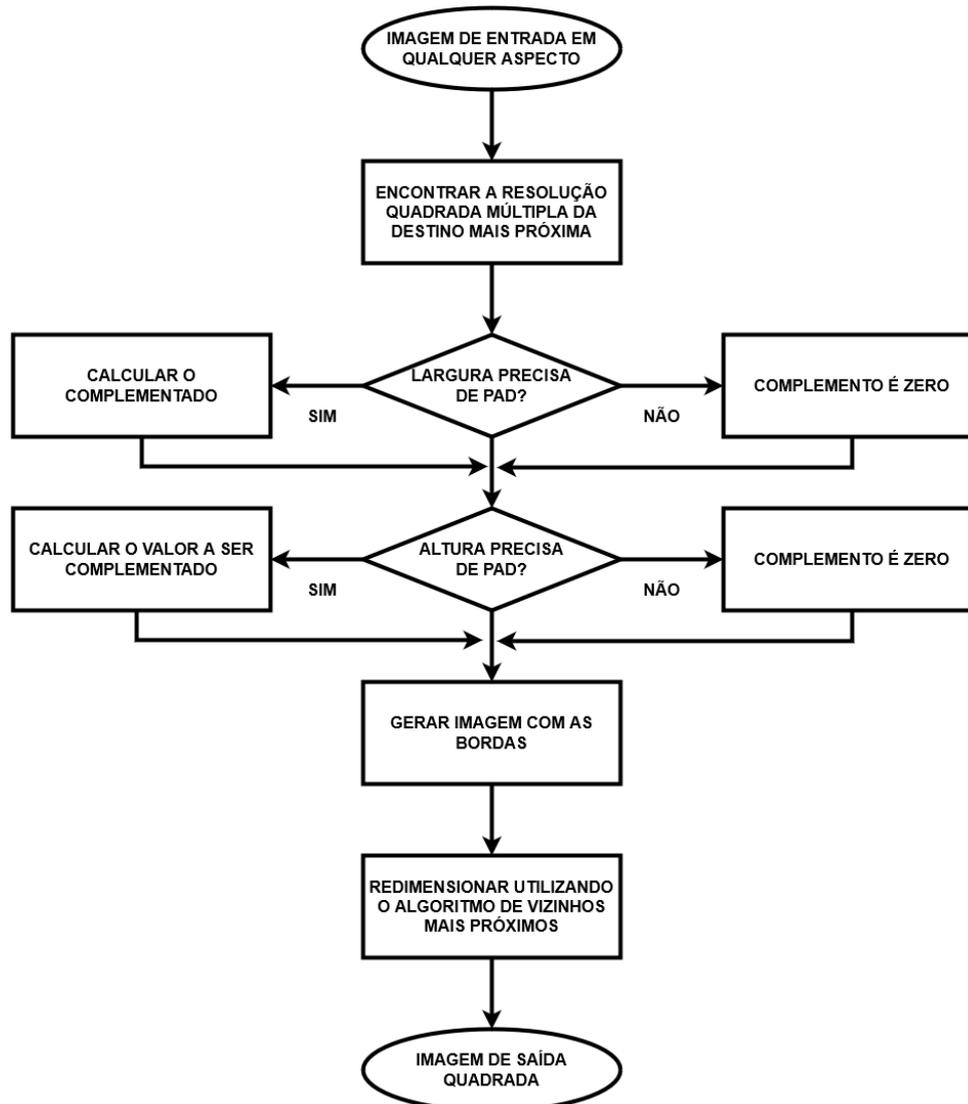
O passo final do algoritmo é aplicar uma operação de redimensionamento com técnica de interpolação de vizinhos mais próximos (*nearest neighbor*). De maneira geral, a literatura sugere a realização da operação de redimensionamento desconsiderando os passos anteriores e operando apenas este redimensionamento da imagem para a resolução de entrada da rede. O código necessário a essa atividade é apresentado no Anexo A.

O resultado deste algoritmo pode ser visto na Figura 18. Na Figura 18a a imagem original é apresentada, na imagem Figura 18b a imagem redimensionada sem a conservação do aspecto original (diretamente) e na Figura 18c o método proposto em Ribeiro e Regis (2022) (*padded*) é aplicado. As imagens foram redimensionadas para serem inseridas no texto.

Similar ao que pode ser visto nas Figuras 18a, 18b e 18c para a imagem 15 do banco DRDC2015, tem-se as Figuras 18d, 18e e 18f e as Figuras 18g, 18h e 18i, para as imagens 17 e 99, respectivamente, do mesmo banco.

Para as imagens da Figura 18 foi calculada a SI e seus valores são apresentados na Tabela 4. Analisando a Tabela 4 é possível inferir que o redimensionamento gera

Figura 17 – Diagrama em blocos da solução proposta para redimensionamento.

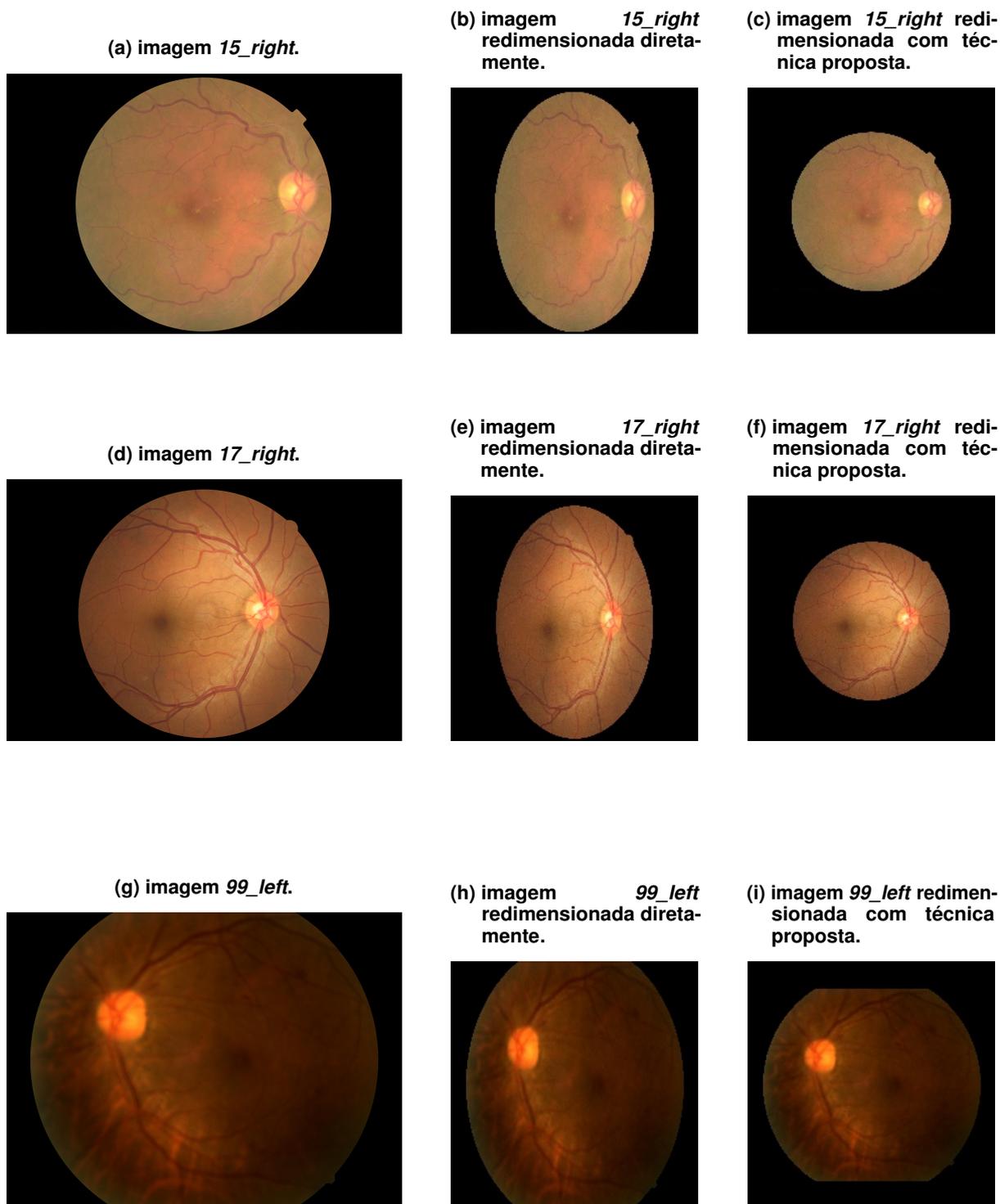


Fonte: Ribeiro e Regis (2022, p. 5).

artefatos, visto que o ajuste sem os cuidados sugeridos podem levar a rede a aprender informações que foram inseridas no processo de redimensionamento, sendo esperada essa inserção de informações apenas na etapa de aumento de dados.

A disparidade entre as imagens Figura 18b e Figura 18c também pode ser calculada por meio de métricas, como o Relação Sinal-Ruído de Pico (*Peak Signal to Noise Ratio* - PSNR) e a métrica de similaridade estrutural (*Structural Similarity* - SSIM), que são técnicas objetivas de avaliação de similaridade (HORE; ZIOU, 2010). O PSNR obtido foi de 14,35, enquanto que o SSIM foi de 0,65, considerando que o PSNR tende ao infinito quando as imagens são idênticas (HORE; ZIOU, 2010) e analisando em conjunto com a medida de SSIM que tende a 1 para imagens idênticas (HORE; ZIOU, 2010).

Figura 18 – Descrição do problema de redimensionamento de imagens de fundoscopia.



As imagens redimensionadas deveriam ser idênticas, porém, tanto pelas métricas de avaliação objetiva, quanto pela SI e pela análise visual, é observável que

Tabela 4 – Informação espacial das imagens da Figura 18.

Arquivo	Descrição	Resolução	SI
15_right	original	3264 × 4928	13,22
	diretamente	224 × 224	67,81
	<i>padded</i>	224 × 224	60,63
17_right	original	2592 × 3888	13,94
	diretamente	224 × 224	50,64
	<i>padded</i>	224 × 224	45,84
99_left	original	1920 × 2560	7,76
	diretamente	224 × 224	25,95
	<i>padded</i>	224 × 224	27,48

a imagem redimensionada diretamente se apresenta de forma muito mais oblonga do que a imagem original. Para as imagens *15_right* e *17_right* percebe-se que a quantidade de informações espaciais inseridas no processo de redimensionamento é inferior quando utilizando a técnica, contudo na imagem *99_left* o processo gerou mais artefatos.

O uso dessa técnica de redimensionamento também permitiu otimizar o aprendizado da máquina, uma vez que as imagens de entrada possuíam resoluções com relação largura/altura distintas, como pode-se verificar na Tabela 5. Percebe-se que em alguns casos a proporção difere. É ressaltado que as resoluções apresentadas nesta tabela são as resoluções prevalecentes, conforme apresentado na Tabela 3, na qual foi detalhado que 21% da base possui resoluções diversas dessas relacionadas.

Tabela 5 – Relação largura/altura para as diferentes resoluções.

Resolução	Relação
3888 × 2592	1,5
4752 × 3168	1,5
3504 × 2336	1,5
2560 × 1920	1,3
2592 × 1944	1,3

4.3 AUMENTO DE DADOS

Após análise do estado da arte, verifica-se que as imagens disponibilizadas são pouco uniformes, variando em diversos aspectos, como luminosidade, proximidade, posicionamento, resolução e etc. Para que os modelos sejam treinados de forma a generalizar o aprendizado, o uso de base de dados com grande quantidade de dados

é sugerido (PEREZ; WANG, 2017).

Algumas técnicas são empregadas com o objetivo de generalizar o aprendizado, mesmo utilizando base de dados consideradas pequenas pela literatura. Dentre elas destaca-se o aumento de dados, adicionar uma camada de regularização, realizar o descarte dos valores treinados para um neurônio (*dropout*), realizar a normalização do lote (*batch normalization*) e adotar a transferência de aprendizado (*transfer learning*) (PEREZ; WANG, 2017).

Como dito, dentre as ferramentas existentes pra diminuir o *overfitting* da rede, um meio importante é ampliar a quantidade de dados disponíveis para treinamento (PEREZ; WANG, 2017). Uma prática de aumento de dados é o emprego de variações dos dados de entrada. Nessa técnica, as imagens da base de dados são submetidas a uma série de processamentos, como espelhamento horizontal e vertical, deslocamento no comprimento ou na altura, recorte, mudança na paleta de cores (PEREZ; WANG, 2017), rotação e zoom, com o objetivo de aumentar a quantidade e diversidade de imagens disponíveis na base.

Por conta de políticas de privacidade relacionadas as imagens, a indústria médica é particularmente beneficiada de um aumento da quantidade de imagens disponíveis (PEREZ; WANG, 2017). O aumento de dados, portanto, se apresenta como um caminho para aumentar a acurácia da classificação (PEREZ; WANG, 2017).

Neste trabalho, após o redimensionamento das imagens e também da divisão da base em treinamento, validação e teste, foram implementados as seguintes formas de aumento de dados no subconjunto de treinamento: espelhamento horizontal, aproximação (*zoom in*) e afastamento (*zoom out*), alteração de brilho, alteração das camadas RGB, deslocamento do centro da imagem na vertical e na horizontal.

O uso dessas técnicas baseia-se nas diferentes características que aparecem dentro da base de dados. Na Figura 19, pode-se ver algumas das imagens presentes no banco de dados e como elas diferem apresentando características similares às emuladas pela técnicas acima descritas.

Nota-se que mesmo no título de cada sub imagem da Figura 19 destacando uma das alterações observadas na imagem em comparação com as outras, ainda é possível perceber que estas alterações podem ser combinadas de maneira “aleatória”.

A técnica de aumento de dados possibilitou que cada uma das imagens de teste gerassem cinco novas, com características como as relacionadas na Tabela 6.

Para a técnica de zoom foi definido que a imagem poderia sofrer uma aproximação ou afastamento aleatório de até 20%. O resultado dessa operação pode ser visto na Figura 20c. A técnica de deslocamento horizontal e vertical, por sua vez, permitiam

Figura 19 – Imagens do conjunto de dados com diferenças exploradas no aumento de dados.

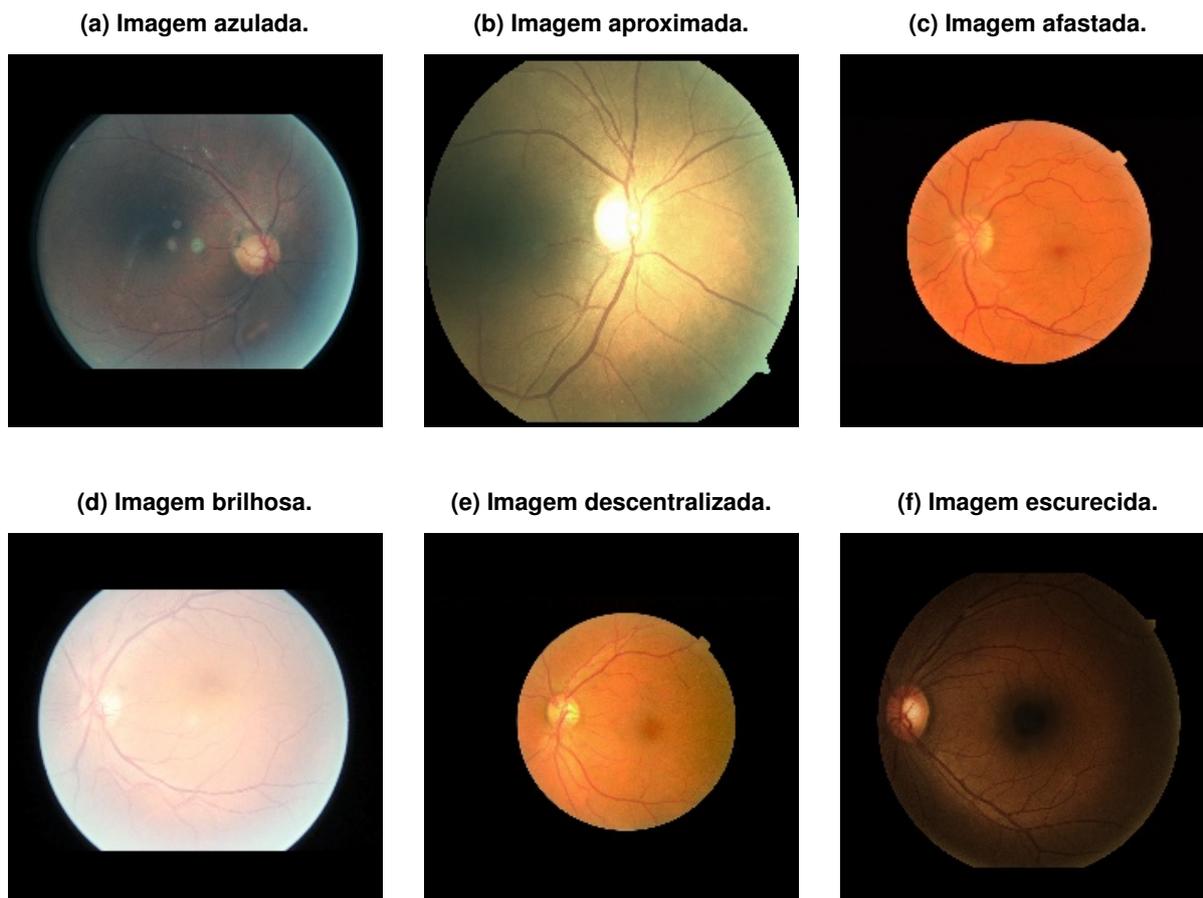


Tabela 6 – Descrição das imagens geradas no aumento de dados.

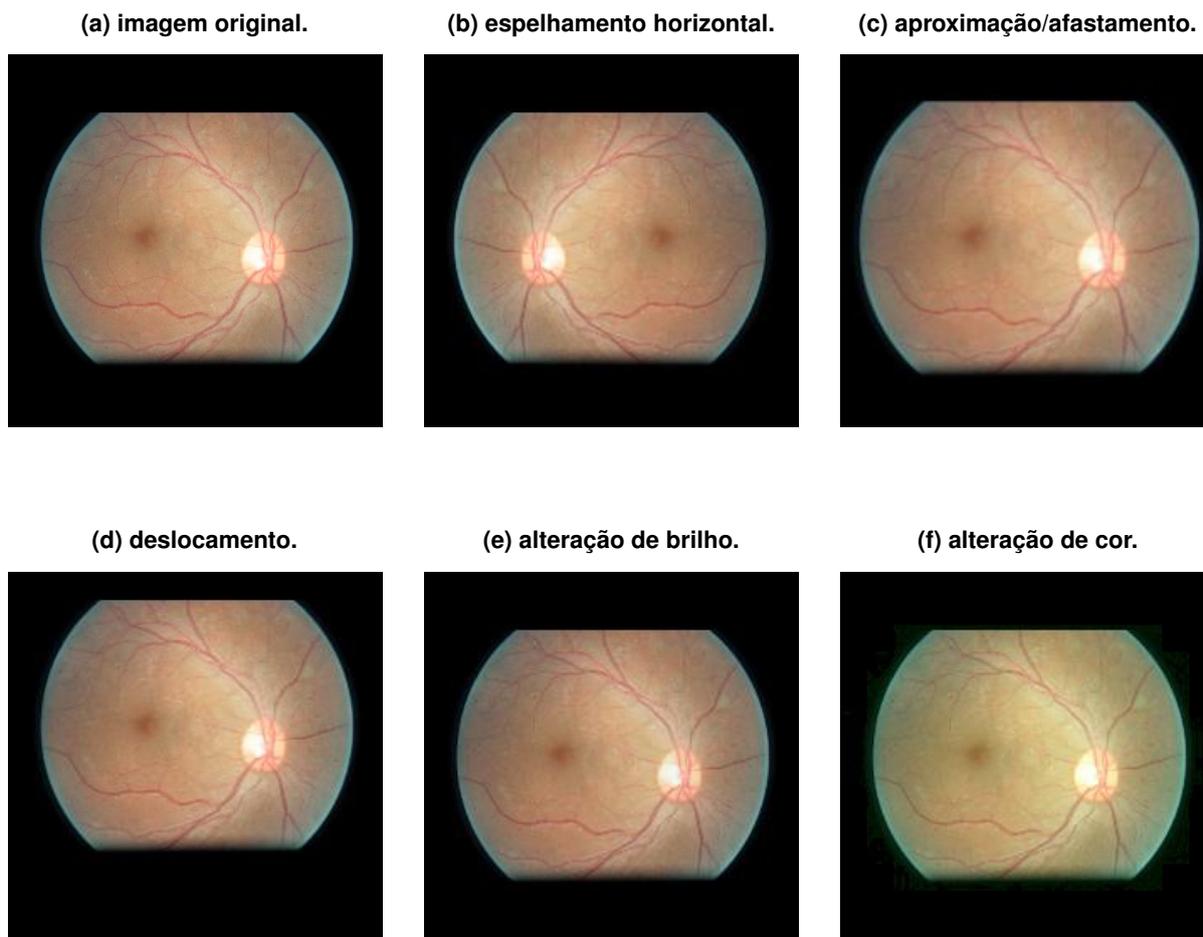
Imagem	Técnica empregada
Figura 20b	Espelhamento horizontal
Figura 20c	Zoom
Figura 20d	Deslocamento horizontal e vertical
Figura 20e	Alteração do brilho
Figura 20f	Alteração de cor

que a imagem fosse deslocada aleatoriamente em até 10% do seu eixo x ou y para ambas as direções. O resultado desta técnica pode ser visto na Figura 20d.

As técnicas de aumento do brilho e de variação de cor modificavam de forma aleatória em até 10% a intensidade dessas características. Seus resultados podem ser vistos, respectivamente, na Figura 20e e na Figura 20f.

As imagens geradas com essa ferramenta em conjunto com as imagens originais

Figura 20 – Exemplos dos aumentos de dados.



utilizadas formaram um conjunto de treinamento com mais de 147 mil imagens, sendo uma quantidade maior do que quatro vezes a base de dados completa. O código para a geração dessas imagens é apresentado no Apêndice A.

4.4 REDE NEURAL

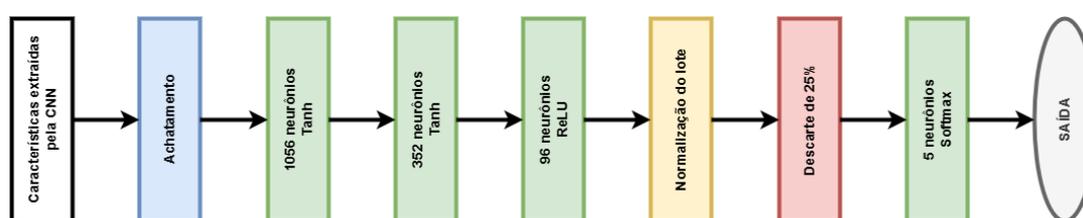
Neste trabalho foram utilizadas as rotulações da base como apresentadas na Tabela 2. Desta forma, a tarefa a ser realizada é a classificação da patologia e das suas graduações. Como forma de compensar o desbalanceamento da base de dados, foi empregada uma técnica de *downsampling*, onde as imagens, depois de aumentadas, foram descartadas do treinamento para que se adequassem à quantidade da classe menos presente na base.

Nas CNN a quantidade de características extraídas depende da arquitetura, da resolução de entrada e se a imagem é colorida ou em escala de cinza, de forma

que a correta escolha sobre a resolução a ser empregada na rede estabelece uma relação de custo/benefício, onde o custo é o tempo necessário ao treinamento da rede, e o benefício é a taxa de acurácia que pode ser obtida nesta resolução. Alguns trabalhos como o de Pires *et al.* (2019) adotam o uso de diferentes resoluções e uma combinação destas redes treinadas com as diferentes resoluções.

As redes neurais utilizadas neste trabalho tiveram como etapa de extração de *features* uma arquitetura CNN e como etapa de classificação, uma rede *multilayer perceptron* com configuração idêntica, de forma a avaliar de forma isométrica as arquiteturas.

Figura 21 – Diagrama da etapa MLP utilizada na rede neural.



Em uma primeira análise, a técnica de aumento de dados descrita foi descartada para verificar os efeitos do processo de redimensionamento relatado anteriormente, porém, no desenvolvimento do trabalho, optou-se por utilizar deste recurso visando permitir uma maior generalização do modelo, haja vista que o emprego desse tipo de estratégia não altera a classificação da imagem, mas é capaz de criar um novo conjunto de imagens a partir de uma única imagem.

Durante os testes realizados neste trabalho, foram empregadas técnicas de otimização de hiper-parâmetros. Nesta etapa foi utilizado o método de busca aleatória (*Random Search*) dos hiper-parâmetros relacionados na Tabela 7, onde também são apresentados os intervalos de valores ou as possibilidades, bem como os valores ótimos obtidos durante os ensaios.

Na Tabela 7, o termo unidades diz respeito à quantidade de neurônios na camada densa, e os níveis se referem à proximidade com a etapa convolucional da rede, sendo o nível 1 o mais próximo. O termo ReLU indica a função de ativação “unidade linear retificada” (*Rectified Linear Unit*) e por sua vez, o termo Tanh trata-se da função trigonométrica “tangente hiperbólica”. O sumário da rede implementada pode ser visto em Apêndice D.

Também foi utilizada uma chamada para interrupção de treinamento (*callback*), caso o modelo não apresentasse evolução por mais de dez épocas, ou não houvesse variação superior a 0.01 nesse intervalo na acurácia do modelo.

Uma etapa de ajuste fino foi realizada por mais 20 épocas após o treinamento

Tabela 7 – Descrição dos hiper-parâmetros para busca aleatória.

Hiper-parâmetro	Valores possíveis	Valor ótimo
Unidades no nível 1	512 a 2048	1056
Função de ativação nível 1	ReLU e Tanh	Tanh
Unidades no nível 2	256 a 1024	352
Função de ativação nível 2	ReLU e Tanh	Tanh
Unidades no nível 3	32 a 512	96
Função de ativação nível 3	ReLU e Tanh	ReLU
<i>Dropout</i>	0.0, 0.1, 0.15, 0.2, 0.25 e 0.3	0.25
Taxa de aprendizado	1e-4 a 1e-2	0.00027
Quantidade de épocas	20, 25, 30, 40 e 55	55
Tamanho do lote	8, 16, 24 e 32	24

base, onde foram ajustados os pesos apenas do último terço das camadas do modelo. As demais camadas foram bloqueadas, com uma taxa de aprendizagem de um décimo da utilizada no treinamento principal. As rotinas de *callback* foram mantidas de forma idêntica para essa etapa.

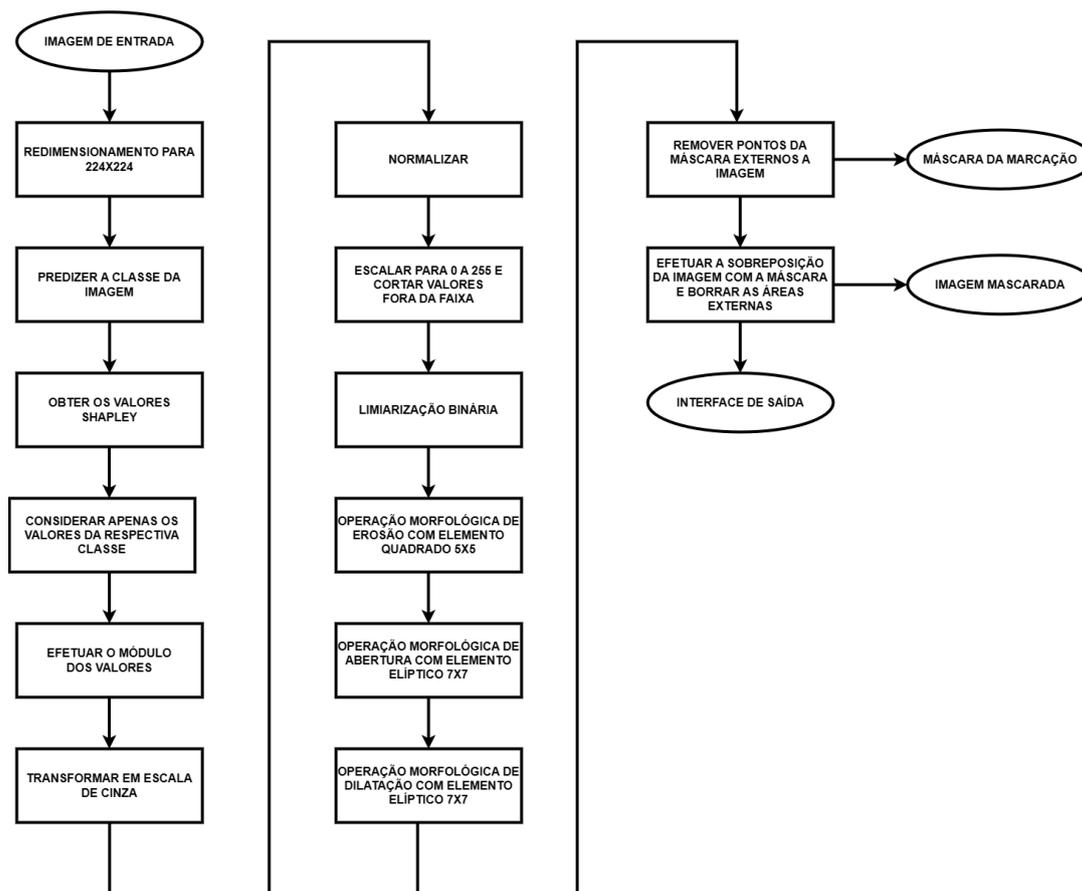
4.5 MÉTODOS DE INTERPRETAÇÃO DO MODELO

O modelo de interpretação da rede utilizado neste trabalho utiliza como base os valores Shapley positivos para compor uma máscara que grafe, nas imagens, áreas relevantes ao profissional oftalmologista objetivando um direcionamento ao profissional como forma de acelerar o processo de avaliação. O algoritmo utilizado é apresentado na Figura 22.

O passo inicial para a marcação da patologia proposta neste trabalho é redimensionar a imagem para que ela tenha o mesmo formato utilizado no treinamento da rede utilizada para a classificação do grau da enfermidade. Na sequência é realizada a predição da intensidade da patologia, em seguida são obtidos os valores Shapley para a imagem. A partir desse passo serão considerados apenas os valores relacionados com a classe estimada, essas ponderações, como descritas anteriormente apontam as áreas mais ou menos relevantes na decisão, desta forma são vistos em módulo para obter as áreas relevantes para a classe, sejam para considerar ou descartar aquele grau.

O passo seguinte consiste em considerar a imagem, que normalmente está em rosado e azul, como escala de cinza utilizando a Equação 10, onde EC é o valor do *pixel* na nova escala, e R , G e B são respectivamente as proporções vermelha, verde e azul do *pixel* na escala de cor original; seguido de uma normalização, escalonamento

Figura 22 – Diagrama em blocos da solução proposta para marcação da área representativa da patologia.



e corte para limitar os valores dos *pixels* entre 0 e 255.

$$EC = \frac{1}{3}R + \frac{1}{3}G + \frac{1}{3}B \quad (10)$$

Na sequência, é realizada uma limiarização binária, para que seja obtida uma imagem com *pixels* binários, sem variações de intensidade entre o preto e o branco. Com a imagem obtida, a operação morfológica erosão com *kernel* em formato de quadrado de tamanho 5×5 é aplicado; posteriormente é operada uma abertura com elemento elíptico 7×7 , finalizando com uma operação morfológica de dilatação com o mesmo *kernel* da operação anterior.

Os pontos externos a fundoscopia então são descartados, obtendo neste ponto a máscara final do algoritmo. O procedimento então opera com a imagem original e a máscara obtida para imprimir a imagem mascarada com um contorno na região e um borramento nas áreas desconsideradas pela máscara, essa imagem também é retornada no algoritmo para finalizar o algoritmo compondo a interface gráfica proposta neste trabalho.

5. RESULTADOS EXPERIMENTAIS

Neste capítulo apresentam-se os resultados obtidos na tarefa de classificação da patologia. Os resultados são discutidos por meio de métricas objetivas, incluindo acurácia, área sob a curva, revocação e precisão, baseados na comparação com a rotulação existente na base de dados DRDC2015. Também é apresentado os resultados para a máscara patológica proposta, comparando com informações disponíveis no conjunto de dados DiaretDB1 de forma subjetiva e objetiva.

Para obter os resultados de validação apresentados neste estudo, foram realizadas 20 iterações para cada arquitetura de rede neural. Durante as iterações, registrou-se apenas o valor médio entre classes (graus da patologia) de cada medida avaliada (ACC, AUC, Revocação e Precisão). Após a conclusão dessas iterações, os dados foram tratados e qualquer iteração com variações superiores a 25%, em relação a média, foi excluída. Em seguida, calculou-se o desvio padrão (σ) e a média aritmética das rodadas remanescentes de cada rede para obter as métricas finais, levando em consideração os valores médios entre classes. Em todas as iterações, utilizou-se o mesmo conjunto de treinamento e validação, garantindo que cada rodada do algoritmo tivesse as mesmas oportunidades de aprendizado e não fosse exposta a conjuntos de dados diferentes.

5.1 CLASSIFICAÇÃO DA PATOLOGIA

As redes receberam imagens com diferentes níveis da patologia e os resultados de cada interação foram calculados como a média desses valores por classe. Assim, a Tabela 8 mostra a média e o desvio padrão obtidos para cada arquitetura testada, considerando as medidas de acurácia (*Accuracy - ACC*), da área sob a curva (AUC), da revocação e da precisão na fase de validação. A média indica o desempenho geral de cada arquitetura, enquanto o desvio padrão fornece uma informação relevante sobre o processo de extração de características entre as simulações.

Tabela 8 – Resultados das métricas de validação para as diferentes arquiteturas.

Arquitetura	Estatística	ACC	AUC	Revocação	Precisão
Densenet121	Média	73,90%	88,16%	73,41%	74,33%
	Desvio Padrão	2,67%	2,82%	2,64%	2,91%
Resnet50	Média	73,54%	86,28%	70,85%	74,82%
	Desvio Padrão	0,20%	3,31%	8,48%	2,90%
Xception	Média	70,82%	88,35%	69,70%	71,60%
	Desvio Padrão	3,15%	1,89%	4,15%	3,37%

Pode-se verificar que o melhor resultado para cada uma das diferentes medidas variou de acordo com a arquitetura. Para a acurácia e para a revocação, a melhor arquitetura foi a Densenet121, enquanto a melhor precisão foi obtida com a arquitetura Resnet50. A medida de AUC, contudo, foi superior para a Xception, merecendo destaque que essa medida representa um equilíbrio entre revocação e precisão.

A diferença de acurácia mais perceptível ocorreu com a arquitetura Xception, que quando comparada as outras arquiteturas, teve uma disparidade de quase três pontos percentuais abaixo das outras arquiteturas.

Os resultados da medida de AUC mostram-se aproximados para as redes Densenet121 e Xception, sendo superiores para a segunda. A revocação, que é uma medida que detecta as classificações positivas, apresenta as maiores diferenças, sendo o pior resultado obtido pela Xception.

A métrica de precisão, que reflete a confiabilidade do modelo ao classificar como positivo, ficou entre 71,6% e 74,8% para todas as arquiteturas, apresentando valores similares para a Densenet121 e Resnet50.

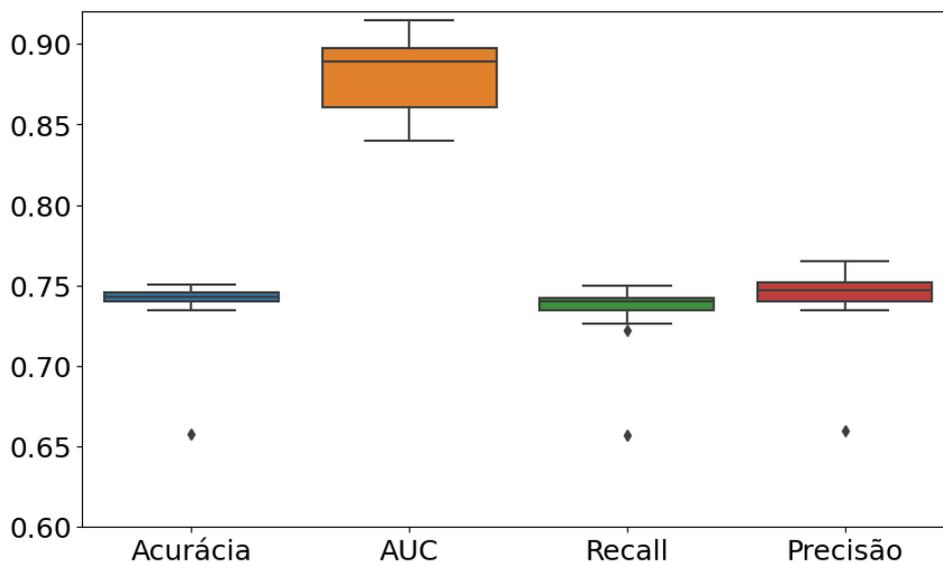
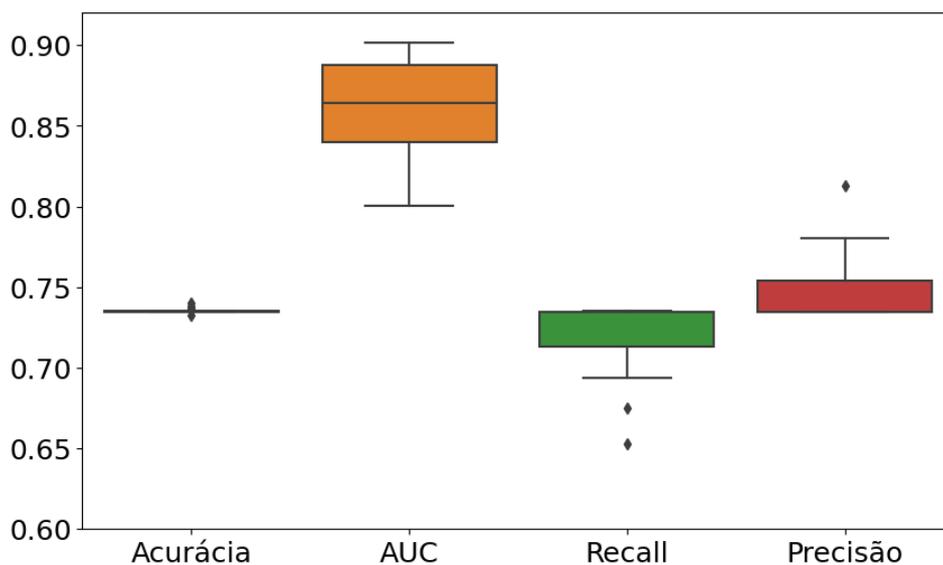
Ao analisar a medida do desvio padrão para os testes, é perceptível que os resultados obtidos entre as iterações do algoritmo são aproximados umas das outras, alcançando-se o menor desvio para a medida de acurácia com a rede Resnet50. Enquanto que o maior desvio para esta métrica foi obtida com a arquitetura Xception.

A variação das métricas de avaliação da rede com arquitetura Densenet121 pode ser vista na Figura 23. Percebe-se que a acurácia, bem como a revocação e a precisão, apresentaram um ponto que difere das demais observações estatísticas (*outliers*) para os testes e observa-se que as maiores variações foram na medida AUC.

O gráfico de caixas para a arquitetura Resnet50 é retratado na Figura 24. Diferentemente do ocorrido para a Densenet121, aqui os resultados de acurácia são extremamente condensados. Para revocação, a amplitude do intervalo é maior que na rede anterior e estão presentes dois *outliers*. Uma atenção especial é devida na AUC, que apresenta valores com amplitude ainda superior à arquitetura anterior. A precisão, por fim, apresenta uma variação também ampla e um *outlier* com resultado positivo, superior a 80%.

Por fim, a rede Xception tem distribuição dos valores conforme a Figura 25. Nesta arquitetura vê-se que os resultados obtidos para acurácia, revocação e precisão são mais variados que os das demais redes. Sendo essa rede a única a apresentar um *outlier* para a medida de AUC.

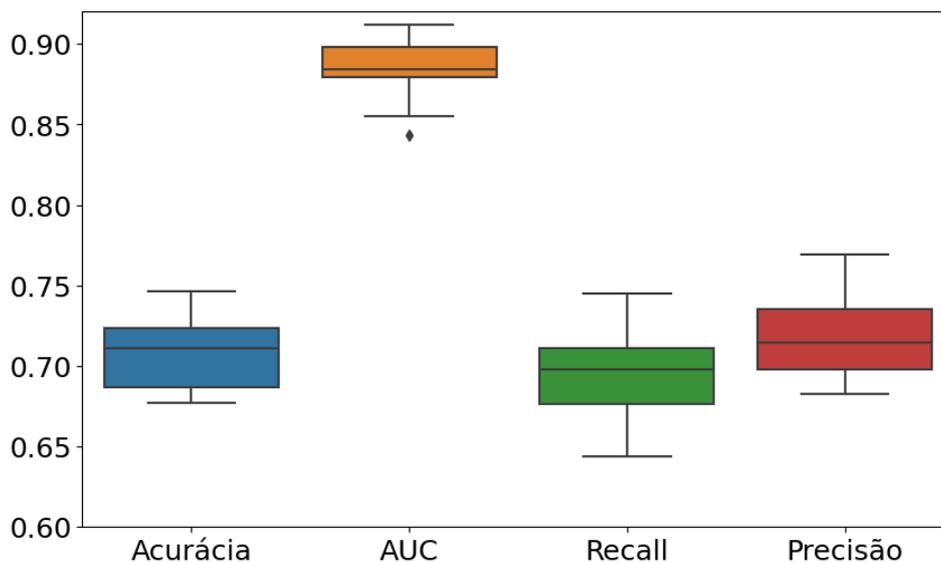
Para conjuntos de dados desbalanceados, como o utilizado neste estudo, a acurácia não se apresenta como uma métrica individual eficiente para avaliar o desem-

Figura 23 – Diagrama de caixas com a distribuição dos resultados das métricas para a rede Densenet121.**Figura 24 – Diagrama de caixas com a distribuição dos resultados das métricas para a rede Resnet50.**

penho, uma vez que tende a estar polarizada pela classe mais presente (QUMMAR *et al.*, 2019). Por este motivo foram avaliadas outras medidas que pudessem complementar e fundamentar a escolha da rede para o passo seguinte, de forma que, em uma análise geral, a Densenet121 se apresentava como a arquitetura mais competente para a classificação dos graus da patologia com base na imagem de entrada.

A melhor arquitetura, então, foi apresentada ao conjunto de testes, onde foram obtidas as medidas expressas na Tabela 9. Por meio da análise destes resultados é possível perceber que o conjunto de validação cumpriu seu papel em avaliar a capacidade de generalização da rede durante a etapa de treinamento. O código para a

Figura 25 – Diagrama de caixas com a distribuição dos resultados das métricas para a rede Xception.



realização dessa tarefa pode ser observado no Apêndice E.

Tabela 9 – Resultado da etapa de teste da classificação dos graus da patologia.

Arquitetura	ACC	AUC	Revocação	Precisão
Densenet121	74,34%	91,32%	72,27%	75,53%

Na Figura 26 são retratadas as curvas de característica de operação do receptor para cada uma das classes diferenciadas neste trabalho (Seção 3.3), bem como a curva média, que representa o resultado expresso na Tabela 9. Destaca-se no gráfico a representação da classe proliferativa, que atingiu AUC na casa dos 93%. Também é observável que a diferenciação de casos sem patologia, leve e moderada apresentam os piores desempenhos.

Verifica-se que a métrica AUC média obtida no melhor caso se assemelha ao trabalho de Pires *et al.* (2019), sendo neste trabalho obtido 91,32% contra os 95,5% da literatura. A acurácia (74,34%) também é comparável à obtida em outros trabalhos (LAM *et al.*, 2018; QUMMAR *et al.*, 2019; GANGWAR; RAVI, 2021), respectivamente 74,5%, 80,8% e 82,2%.

Na Figura 27 é apresentada a matriz de confusão, que mostra o desempenho do algoritmo na detecção da patologia em imagens de fundoscopia. Nesta Figura nota-se que os maiores índices se encontram na diagonal principal da matriz, o que corrobora para a interpretação de que se trata de um resultado adequado.

Figura 26 – Curva ROC dos resultados obtidos.

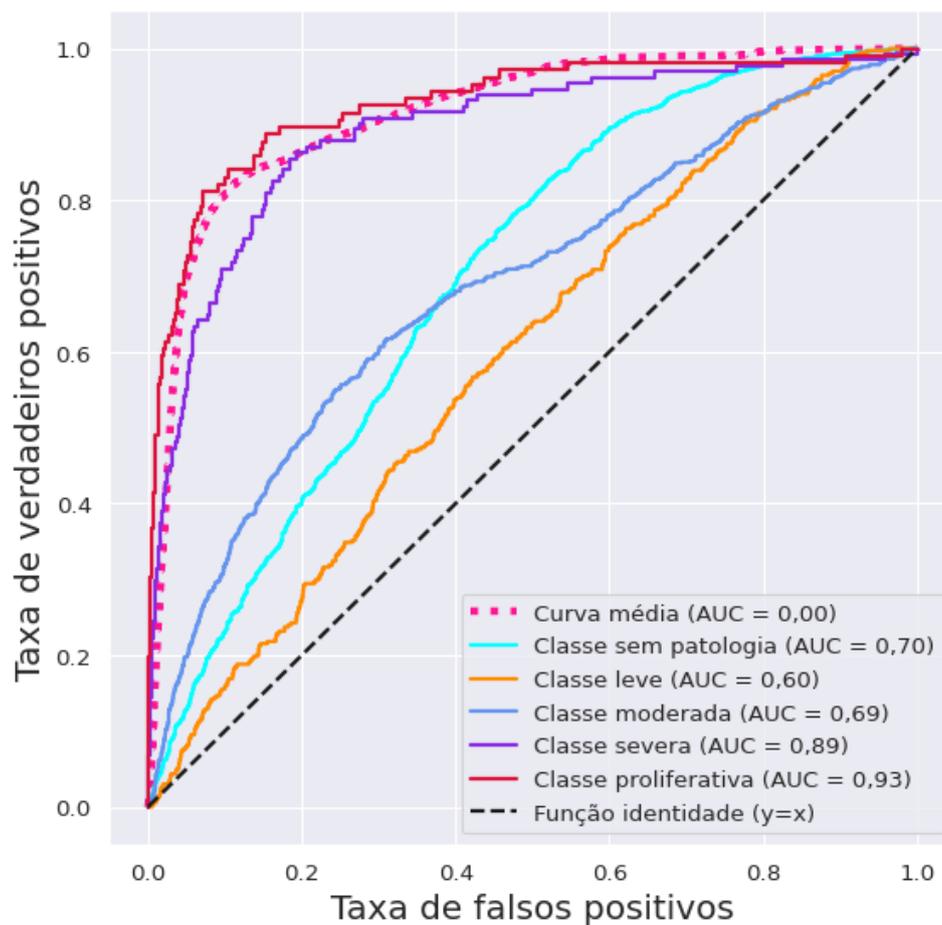
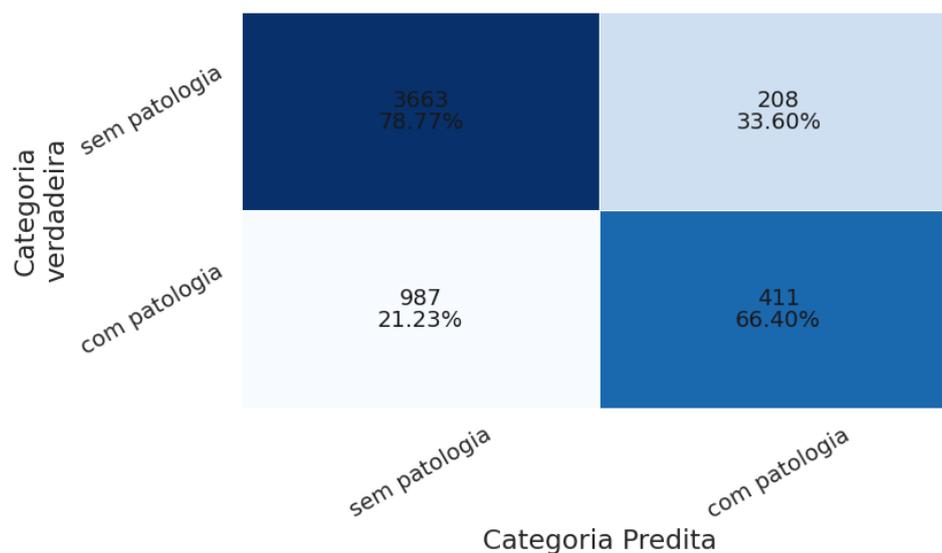


Figura 27 – Matriz de confusão para a etapa de teste da classificação patológica ou não.



5.2 ANÁLISE DA MÁSCARA PATOLÓGICA PROPOSTA

Uma segunda implementação deste trabalho é um método de marcação em imagem das áreas relevantes para o profissional de oftalmologia, nas quais são observáveis microaneurisma, exudatos duros ou algodonosos, e hemorragia em formato de ponto. Estes são os destacados e comparáveis com a base de dados DiaretDB1, mas não são o todo, já que o algoritmo foi apresentado a imagens também com hemorragias diversas (ponto, chama), alterações nos vasos (perolização venosa, alteração dos caminhos), entre outros indícios.

A arquitetura que apresentou o melhor resultado para classificação da intensidade da patologia foi utilizada em conjunto com o algoritmo proposto no Capítulo 4 (Figura 22). Na imagem da esquerda da Figura 28, é apresentada a imagem de entrada após redimensionamento; a do meio traz a máscara obtida pelo algoritmo e na imagem da direita é mostrada a imagem sobreposta destacando as áreas indicadas como patológicas da imagem. Na Figura 28 também é relatada a predição de qual o grau da patologia presente na imagem.

Figura 28 – Saída do algoritmo.



5.2.1 Análise subjetiva

Como forma de comparar de maneira qualitativa os resultados do algoritmo proposto, conforme descrito no Capítulo 4, foram usadas imagens da base DiaretDB1. Onde para a mesma imagem encontra-se a imagem apresentada em Figura 28 e as marcações para os diferentes sinais da patologia, como visto na Figura 29.

É perceptível por meio da comparação entre Figura 28 e Figura 29 que as áreas marcadas pelo algoritmo se assemelham em parte às áreas marcadas pelos profissionais.

Na Figura 30 as máscaras das Figuras 29b a 29e são sobrepostas para representar as áreas patológicas da imagem. Assim, foi obtida uma imagem com a

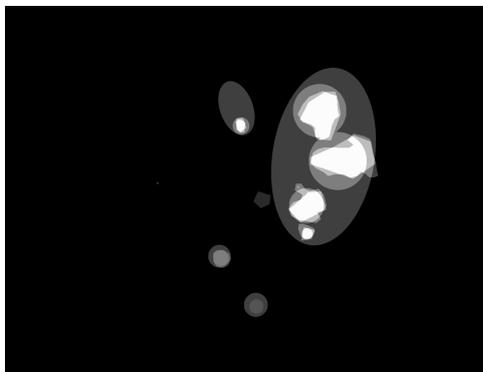
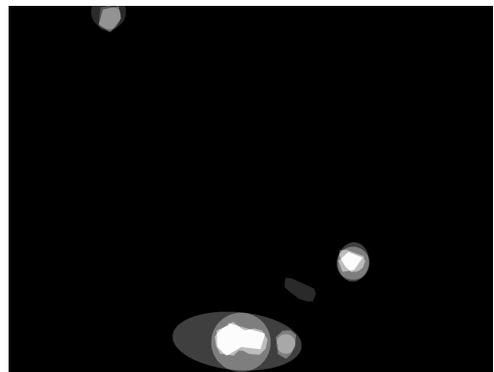
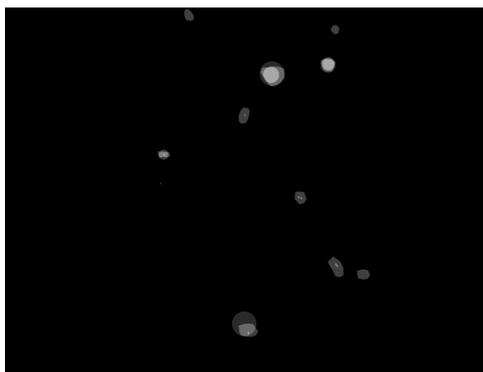
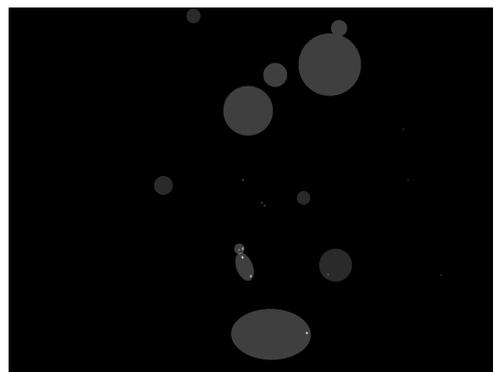
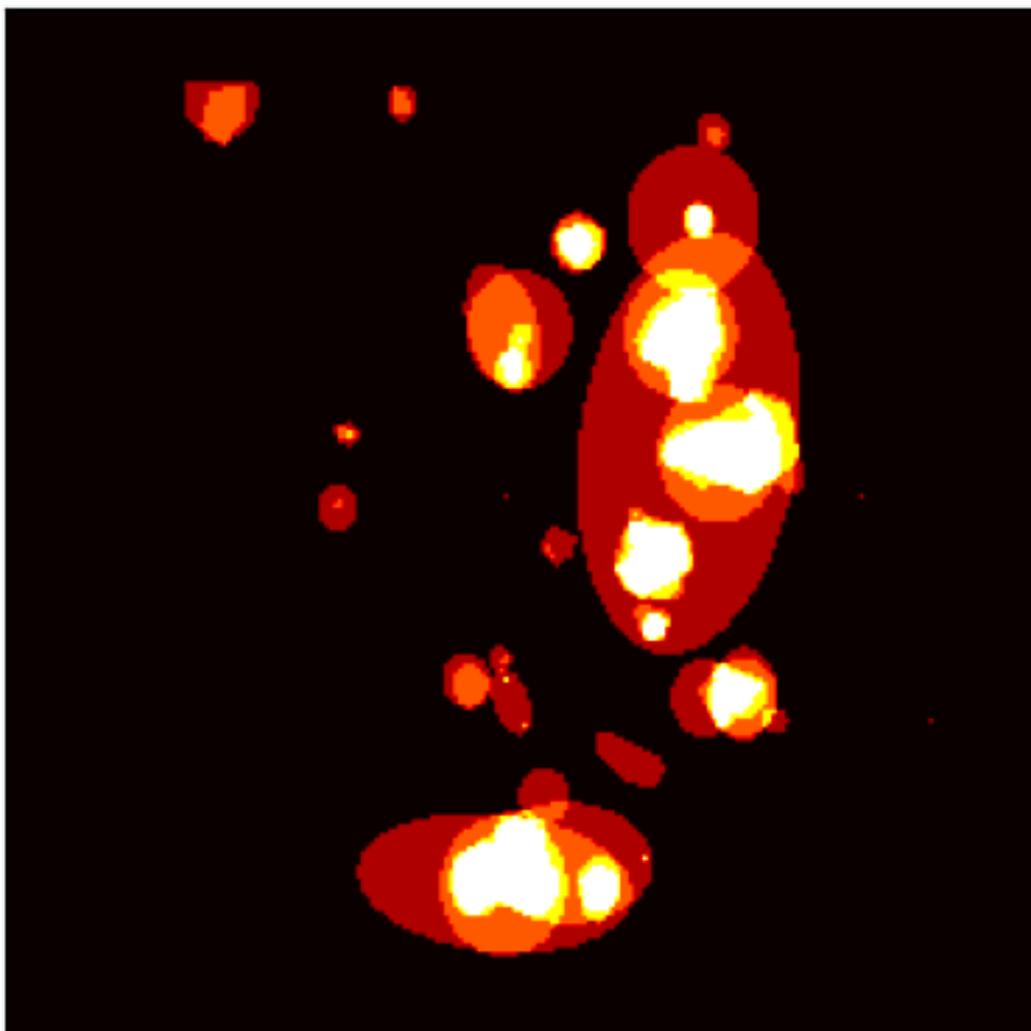
Figura 29 – Imagem grafada por profissional.**(a) imagem original.****(b) marcação dos exudatos duros.****(c) marcação dos exudatos algodonosos.****(d) marcação das hemorragias.****(e) marcação dos aneurismas em pontos.**

Imagem 8 disponível no DiaretDB1 (KÄLVIÄINEN; UUSITALO, 2007).

representação de todas as regiões de enfermidade grafadas com intensidades variando de um a quatro a depender da quantidade de profissionais que tiverem efetuado a marcação da localidade.

Figura 30 – Sobreposição das marcações patológicas.

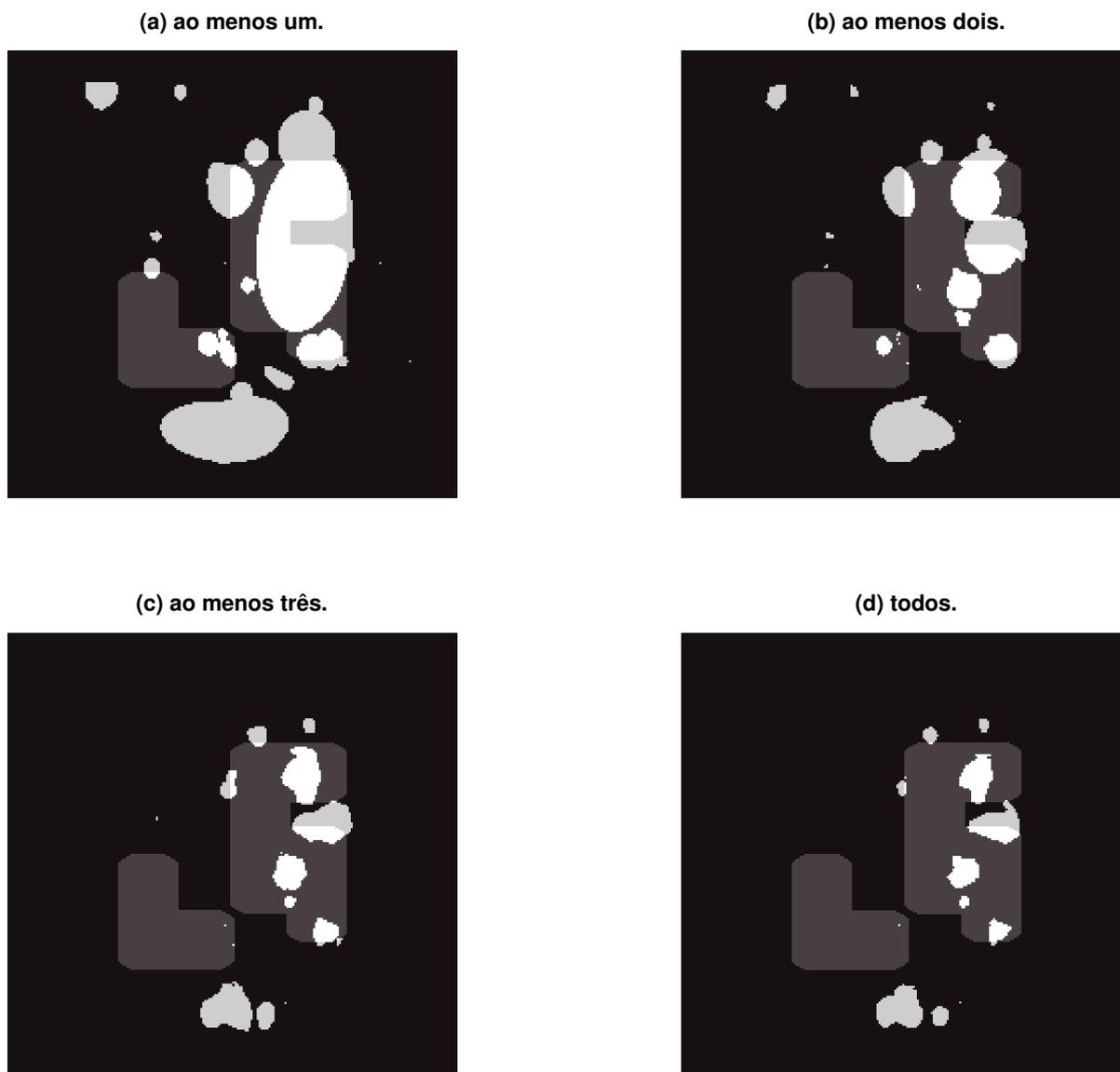
A imagem resultante da Figura 30 é então dividida pelo grau de intensidade de forma a classificar áreas pela quantidade de profissionais que indicaram a patologia, definindo onde ao menos um, dois, três ou todos os profissionais marcaram, respectivamente representadas nas Figuras 31a, 31b, 31c e 31d.

Analisando a Figura 31 pode-se observar que as áreas sobrepostas são extensas, principalmente para Figura 31a e Figura 31b. Subjetivamente pode-se afirmar que a marcação mesmo que necessitando refinamento, já se mostra relevante por conseguir indicar áreas onde são observados sinais da patologia.

5.2.2 Análise objetiva

Almejando avaliar as marcações propostas pelo algoritmo, foi utilizada a métrica de acurácia, em conjunto com a medida de IoU, definido na Equação 8. Para essa comparação ser viável, foi necessário realizar a binarização da imagem, de forma a

Figura 31 – Sobreposição da máscara do algoritmo para diferentes quantidades de marcações profissionais.



comparar a máscara proposta com a máscara profissional em um dado limiar.

Como dito, o IoU é um coeficiente que avalia a sobreposição entre máscaras, onde a máscara predita e a máscara obtida pela binarização do somatório das marcações de todos os indícios patológicos disponíveis no DiaretDB1, em que todos os profissionais sinalizaram a existência de patologia, são comparador.

Na Tabela 10 são apresentados os resultados obtidos para a acurácia e o interseção sobre união proposto para as marcações por quantidade de profissionais

Tabela 10 – Sumários das métricas da marcação.

Imagem	ACC	IoU
Figura 31a	77,62%	0,3306
Figura 31b	81,18%	0,1718
Figura 31c	83,07%	0,1143
Figura 31d	83,62%	0,0875

que identificaram aquelas regiões. A grande diferença nas medidas se justifica pelo fato de a acurácia considerar um acerto quando os *pixels* sem indicativos da patologia não são regiões realçadas, ou seja, representam verdadeiros negativos e estão em preto nas duas máscaras.

Desta forma considera-se que a métrica de acurácia obtida pela marcação do algoritmo, em comparação com a efetuada pelos profissionais, tem uma semelhança considerável, haja vista o valor superior a 80%. Assim sendo, a medida de avaliação numérica IoU aparenta ser mais assertiva na avaliação uma vez que considera apenas a sobreposição dos *pixels* que estão sendo destacados.

6. CONCLUSÃO

Este trabalho apresenta uma técnica de classificação e marcação em imagens de fundoscopia, suportada em uma nova óptica sobre o redimensionamento das imagens, fundamentada nas técnicas de processamento digital de imagens, de forma a preparar essas imagens para serem aplicadas em redes neurais convolucionais. Para tanto, foi proposto também ao longo do trabalho empregar técnicas pertinentes de aumento de dados, conforme análise apresentada, para ampliar as capacidades de generalização do modelo, que vão suportar a tarefa de detecção e graduação da retinopatia diabética. Por fim, foi ainda utilizada uma estratégia de marcação das áreas patológicas por meio da ponderação Shapley. Desta forma obtendo um *script* capaz de auxiliar no diagnóstico da patologia que se propôs analisar.

Como visto ao longo da pesquisa e abordado no Capítulo 5, as diferentes arquiteturas de redes convolucionais empregadas neste trabalho não apresentaram diferenças expressivas nas métricas de avaliação da tarefa de classificação.

Durante as simulações foi possível perceber que as arquiteturas de redes neurais convolucionais utilizadas neste trabalho apresentam, mesmo que parcialmente, robustez e capacidade de extrair características relevantes para a graduação da patologia, ou seja, que compreendam as diferenças, entre os padrões patológicos, sendo algumas das classes mais difíceis na diferenciação.

A análise da região do olho com sinais anômalos que esse trabalho se propôs em auxiliar é demonstrada na Figura 28, que deve ser comparada com as informações fornecidas pela base de dados dos especialistas, referentes à mesma imagem na Figura 29. Pode-se perceber que um refinamento no algoritmo proposto é necessário, contudo, em uma análise qualitativa, tanto objetiva quanto subjetiva, as áreas demarcadas se assemelham, em especial para os exudatos duros, sendo as áreas marcadas pelos profissionais aproximadas e/ou vizinhas às obtidas com a técnica desenvolvida neste trabalho.

Como ressaltado o objetivo é desenvolver uma técnica que auxilie o profissional na tarefa de diagnosticar a RD e não se limitar a encontrar uma rede ou arquitetura de CNN que obtenha métricas superiores as apresentadas na literatura para a tarefa de classificação. Contudo, nos resultados finais deste trabalho, comparou-se com êxito os dados obtidos com os valores apresentados em Wan, Liang e Zhang (2018), para AUC se assemelha ao trabalho de Pires *et al.* (2019), enquanto que a acurácia é comparável à obtida nos trabalhos de Lam *et al.* (2018), Qummar *et al.* (2019), Gangwar e Ravi (2021).

Merece também ênfase que diferentes profissionais realizam a marcação de

diferentes áreas como mostrado na Figura 29, alguns com mais ou menos refinamento quando comparados uns aos outros, o que reforça a relevância da marcação do *script*, com a arquitetura Densenet121, para a tarefa proposta, sendo reafirmado pelos resultados sumarizados na Tabela 10.

Salienta-se ainda que os recursos computacionais para operar com esta ampla quantidade de imagens, mesmo que em uma resolução proporcionalmente baixa em comparação com as originais da etapa de aquisição, são vastos, o que por várias vezes causou gargalos a serem vencidos durante a pesquisa. Na fase de marcação, ainda foram enfrentados problemas de memória relacionados à obtenção do descritivo do modelo pelo método Shapley, sendo necessários direcionamentos que potencialmente também reduziram a qualidade das marcações finais.

6.1 CONTRIBUIÇÕES

As principais contribuições apresentadas neste trabalho são listadas a seguir:

- Desenvolvimento de um *script* capaz auxiliar no processo de classificação da retinopatia diabética;
- Desenvolvimento de código para geração de imagens com aumento de dados;
- Desenvolvimento de um *script* que permita saída gráfica incluindo marcação da área relacionada a patologia;
- Avaliação da rede de classificação desenvolvida;
- Proposição de uma métrica objetiva para análise da área patológica;
- Avaliação da marcação realizada pelo algoritmo.

6.2 SUGESTÕES PARA TRABALHOS FUTUROS

- Revisar as rotulações presentes na base DRDC2015 com um especialista;
- Elaborar outras técnicas de aumento de dados, como ruído gaussiano aditivo, desfoque, nitidez, relevo e gama aleatória e seus efeitos no desempenho do algoritmo;
- Verificar a possibilidade de melhor desempenho da marcação pelo emprego de outras técnicas, como redes neurais próprias para segmentação de imagens;
- Comparar o treinamento de uma rede de segmentação de imagens supervisionada com os resultados de marcação do algoritmo proposto;

- Avaliar o desempenho do uso de duas redes, uma que verifique a presença da patologia e outra que gradue a patologia;
- Inserir mais imagens pertencentes às categorias que não foram compreendidas durante o treinamento objetivando melhorar o desempenho da diferenciação entre as classes sem patologia, leve e moderada;
- Buscar outras arquiteturas de rede convolucional;
- Otimizar os algoritmos do ponto de vista do consumo de recursos computacionais e/ou utilizar ambientes com recursos mais robustos;
- Desenvolver uma interface gráfica completa que permita ao profissional inserir a imagem ou efetuar a captura da mesma e apresente o resultado;
- Avaliar a possibilidade de remover o grau leve da patologia, considerando que o este pode se aproximar da ausência de RD, para diminuir os ruídos de treinamento, em especial em imagens de baixa resolução;
- Portar a ferramenta final para uso em dispositivos portáteis como *smartphones* ou *tablets*;
- Realizar o tratamento dos dados das iterações com base em percentil;
- Avaliar o uso de técnicas de redimensionamento de imagens com filtros *anti-aliasing*.

REFERÊNCIAS

- ALBAWI, S.; MOHAMMED, T. A.; AL-ZAWI, S. Understanding of a convolutional neural network. In: IEEE. **2017 international conference on engineering and technology (ICET)**. [S.l.], 2017. p. 1–6.
- BANDELLO, F. *et al.* **Clinical Strategies in the Management of Diabetic Retinopathy: A Step-by-Step Guide for Ophthalmologists**. Springer International Publishing, 2018. ISBN 9783319961576. Disponível em: <https://books.google.com.br/books?id=A_5uDwAAQBAJ>.
- CHEN, Q. *et al.* Choroidal vasculature characteristics based choroid segmentation for enhanced depth imaging optical coherence tomography images. **Medical physics**, Wiley Online Library, v. 43, n. 4, p. 1649–1661, 2016.
- CHOLLET, F. Xception: Deep learning with depthwise separable convolutions. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2017. p. 1251–1258.
- CONGDON, N. G.; FRIEDMAN, D. S.; LIETMAN, T. Important causes of visual impairment in the world today. **Jama**, American Medical Association, v. 290, n. 15, p. 2057–2060, 2003.
- CORRÊA, Z. M. d. S.; JR, R. E. Aspectos patológicos da retinopatia diabética. **Arquivos brasileiros de oftalmologia**, SciELO Brasil, v. 68, p. 410–414, 2005.
- DENG, J. *et al.* Imagenet: A large-scale hierarchical image database. In: IEEE. **2009 IEEE conference on computer vision and pattern recognition**. [S.l.], 2009. p. 248–255.
- EHNES, A. *et al.* Optical coherence tomography (oct) device independent intraretinal layer segmentation. **Translational vision science & technology**, The Association for Research in Vision and Ophthalmology, v. 3, n. 1, p. 1–1, 2014.
- EL-BAZ, A.; SURJI, J. **Diabetes and Fundus OCT**. Elsevier Science, 2020. (Computer-Assisted Diagnosis). ISBN 9780128174418. Disponível em: <<https://books.google.com.br/books?id=D-XEDwAAQBAJ>>.
- EL-BAZ, A.; SURJI, J. **Diabetes and Retinopathy**. Elsevier Science, 2020. (Computer-Assisted Diagnosis). ISBN 9780128174395. Disponível em: <<https://books.google.com.br/books?id=E-XEDwAAQBAJ>>.
- FEDERATION, I. D. **IDF Diabetes Atlas**. Brussels, Belgium: [s.n.], 2021. <<https://www.diabetesatlas.org>>.
- GANGWAR, A. K.; RAVI, V. Diabetic retinopathy detection using transfer learning and deep learning. In: **Evolution in Computational Intelligence**. [S.l.]: Springer, 2021. p. 679–689.
- GOH, J. K. H. *et al.* Retinal imaging techniques for diabetic retinopathy screening. **Journal of diabetes science and technology**, SAGE Publications Sage CA: Los Angeles, CA, v. 10, n. 2, p. 282–294, 2016.

- GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing**. 3. ed. New Jersey: Pearson Prentice Hall, 2008.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep learning**. [S.l.]: MIT press, 2016.
- HAYKIN, S. **Neural Networks: A Comprehensive Foundation**. 2nd. ed. Ontario: Prentice Hall, 1999.
- HAYKIN, S. O. **Neural Networks and Learning Machines**. 3rd. ed. Ontario: Pentrice Hall, 2008.
- HE, K. *et al.* Deep residual learning for image recognition. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2016. p. 770–778.
- HORE, A.; ZIOU, D. Image quality metrics: Psnr vs. ssim. In: IEEE. **2010 20th international conference on pattern recognition**. [S.l.], 2010. p. 2366–2369.
- HUANG, G. *et al.* Densely connected convolutional networks. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2017. p. 4700–4708.
- JAYA, T.; DHEEBA, J.; SINGH, N. A. Detection of hard exudates in colour fundus images using fuzzy support vector machine-based expert system. **Journal of Digital Imaging**, Springer, v. 28, n. 6, p. 761–768, 2015.
- JOSHI, S. *et al.* Analysis of preprocessing techniques, keras tuner, and transfer learning on cloud street image data. In: IEEE. **2021 IEEE International Conference on Big Data (Big Data)**. [S.l.], 2021. p. 4165–4168.
- KAFIEH, R. *et al.* Intra-retinal layer segmentation of 3d optical coherence tomography using coarse grained diffusion map. **Medical image analysis**, Elsevier, v. 17, n. 8, p. 907–928, 2013.
- KÄLVIÄINEN, R.; UUSITALO, H. Diaretdb1 diabetic retinopathy database and evaluation protocol. In: CITESEER. **Medical image understanding and analysis**. [S.l.], 2007. v. 2007, p. 61.
- LAM, C. *et al.* Automated detection of diabetic retinopathy using deep learning. **AMIA summits on translational science proceedings**, American Medical Informatics Association, v. 2018, p. 147, 2018.
- LECUN, Y.; BENGIO, Y. *et al.* Convolutional networks for images, speech, and time series. **The handbook of brain theory and neural networks**, Citeseer, v. 3361, n. 10, p. 1995, 1995.
- LI, Z. *et al.* A survey of convolutional neural networks: analysis, applications, and prospects. **IEEE transactions on neural networks and learning systems**, IEEE, 2021.
- LUNDBERG, S. M.; ERION, G. G.; LEE, S.-I. Consistent individualized feature attribution for tree ensembles. **arXiv preprint arXiv:1802.03888**, 2018.

- LUNDBERG, S. M.; LEE, S.-I. A unified approach to interpreting model predictions. **Advances in neural information processing systems**, v. 30, 2017.
- MOOKIAH, M. R. K. *et al.* Computer-aided diagnosis of diabetic retinopathy: A review. **Computers in biology and medicine**, Elsevier, v. 43, n. 12, p. 2136–2155, 2013.
- OSAADON, P. *et al.* A review of anti-vegf agents for proliferative diabetic retinopathy. **Eye**, Nature Publishing Group, v. 28, n. 5, p. 510–520, 2014.
- OTTAIANO, J. A. A. *et al.* **As Condições de Saúde Ocular no Brasil 2019**. São Paulo, Brasil: [s.n.], 2019. <https://www.cbo.com.br/novo/publicacoes/condicoes_saude_ocular_brasil2019.pdf>.
- PEREZ, L.; WANG, J. The effectiveness of data augmentation in image classification using deep learning. **arXiv preprint arXiv:1712.04621**, 2017.
- PIRES, R. *et al.* A data-driven approach to referable diabetic retinopathy detection. **Artificial intelligence in medicine**, Elsevier, v. 96, p. 93–106, 2019.
- QUMMAR, S. *et al.* A deep learning ensemble approach for diabetic retinopathy detection. **IEEE Access**, IEEE, v. 7, p. 150530–150539, 2019.
- RAKOTOMAMONJY, A. Optimizing area under roc curve with svms. In: **ROCAI**. [S.l.: s.n.], 2004. p. 71–80.
- REGIS, C. D. M. **Métrica de avaliação objetiva de vídeo usando a informação espacial, a temporal e a disparidade**. Tese (Doutorado) — Universidade Federal de Campina Grande, Campina Grande, 2013.
- RIBEIRO, F. R. G.; REGIS, C. D. M. Proposta de redimensionamento de imagens de fundoscopia utilizada para classificar a retinopatia diabética. **XII Conferência Nacional em Comunicações, Redes e Segurança da Informação (ENCOM 2022)**, 2022.
- ROSSANT, F. *et al.* Automated segmentation of retinal layers in oct imaging and derived ophthalmic measures. In: IEEE. **2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro**. [S.l.], 2009. p. 1370–1373.
- SCANLON, P. *et al.* **A practical manual of diabetic retinopathy management**. New Jersey, USA: John Wiley & Sons, 2009.
- SHALEV-SHWARTZ, S.; BEN-DAVID, S. **Understanding machine learning: From theory to algorithms**. [S.l.]: Cambridge university press, 2014.
- SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409.1556**, 2014.
- SZEGEDY, C. *et al.* Going deeper with convolutions. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2015. p. 1–9.
- SZEGEDY, C. *et al.* Rethinking the inception architecture for computer vision. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2016. p. 2818–2826.

TYMCHENKO, B.; MARCHENKO, P.; SPODARETS, D. Deep learning approach to diabetic retinopathy detection. **arXiv preprint arXiv:2003.02261**, 2020.

WAN, S.; LIANG, Y.; ZHANG, Y. Deep convolutional neural networks for diabetic retinopathy detection by image classification. **Computers & Electrical Engineering**, Elsevier, v. 72, p. 274–282, 2018.

YAZDANPANA, A. *et al.* Segmentation of intra-retinal layers from optical coherence tomography images using an active contour approach. **IEEE transactions on medical imaging**, IEEE, v. 30, n. 2, p. 484–496, 2010.

ZHAO, W. *et al.* Shap values for explaining cnn-based text classification models. **arXiv preprint arXiv:2008.11825**, 2020.

ZIA, F. *et al.* A multilevel deep feature selection framework for diabetic retinopathy image classification. TECH SCIENCE PRESS, 2022.

APÊNDICE A – CÓDIGO PARA AUMENTO DE DADOS

```

1 #(...)
2 n_files = 35126
3 df_labels = pd.read_csv('train_filenames.csv')
4 file_dir = 'resized/train'
5 save_dir = 'resized/aumented'
6
7 def rgb2gray(rgb):
8     """
9     Convert an RGB image to grayscale.
10
11     This function takes an RGB image represented as a numpy array and
12     returns a grayscale version of the image using the dot product of
13     the image and a weighting matrix.
14
15     Parameters:
16     rgb (numpy array): The RGB image to be converted to grayscale.
17
18     Returns:
19     numpy array: The grayscale version of the input image.
20     """
21     return np.dot(rgb[...,:3], [0.2989, 0.5870, 0.1140])
22
23 def rgb_noise(orig):
24     """
25     Add random noise to an RGB image.
26
27     This function takes an RGB image represented as a numpy array and
28     returns a new version of the image with random noise added to it.
29     The amount of noise added can be controlled with the INTENSITY
30     variable. The function will randomly select between 0 and 3
31     channels to add noise to, and the order in which the channels are
32     chosen is also random.
33
34     Parameters:
35     orig (numpy array): The original RGB image to add noise to.
36
37     Returns:
38     numpy array: The input image with random noise added to it.
39     """
40     img = orig.copy().astype('float32')
41     INTENSITY = 0.1
42     max = np.max(img) * INTENSITY
43     mutations = random.randrange(0, 4)
44     order = np.random.permutation(['r', 'g', 'b'])

```

```
45     if mutations == 0:
46         return img
47     noise = np.zeros(img.shape)
48     mask = np.where(rgb2gray(img) != 0, 1, 0)
49     for i in range(mutations):
50         if order[i] == 'r':
51             noise[:, :, 0] += max * random.random()
52                 * np.ones((img.shape[0], img.shape[1]))
53                 * mask
54         elif order[i] == 'g':
55             noise[:, :, 1] += max * random.random()
56                 * np.ones((img.shape[0], img.shape[1]))
57                 * mask
58         elif order[i] == 'b':
59             noise[:, :, 2] += max * random.random()
60                 * np.ones((img.shape[0], img.shape[1]))
61                 * mask
62     img += noise.astype('float32')
63     return img
64
65 def generation(trainy, df_labels):
66     trainy_augm = np.ones(6*len(trainy), 'uint8')
67     X_train = df_labels['image']
68     datagen = ImageDataGenerator(fill_mode='constant')
69     fp1 = np.memmap('trainx_with_aumentation.dat',
70                   dtype='float32',
71                   mode='w+',
72                   shape=(6 * len(X_train), 224, 224, 3))
73     for filename, i in zip(X_train, range(0, len(X_train))):
74         img = load_img(f'{file_dir}/{filename}.jpeg',
75                       target_size=(224, 224))
76         img_array = np.array(img) / 255.
77         fp1[6 * i] = img_array
78
79         aumented = datagen.apply_transform(x=img_array,
80                                           transform_parameters={'flip_horizontal':True})
81         matplotlib.image.imsave(f'{save_dir}/{filename}a.jpeg',
82                                 aumented)
83         fp1[6 * i + 1] = aumented
84
85         zoom = 1 + random.choice([-1, 1]) * 0.2 * random.random()
86         aumented = datagen.apply_transform(x=img_array,
87                                           transform_parameters={'zx':zoom, 'zy':zoom})
88         matplotlib.image.imsave(f'{save_dir}/{filename}b.jpeg',
89                                 aumented)
89         fp1[6 * i + 2] = aumented
```

```
91
92     limx = random.choice([-1, 1]) * 0.1
93           * random.random() * img_array.shape[1]
94     limy = random.choice([-1, 1]) * 0.1
95           * random.random() * img_array.shape[2]
96     augmented = datagen.apply_transform(x=img_array,
97                                       transform_parameters={'tx':limx, 'ty':limy})
98     matplotlib.image.imsave(f'{save_dir}/{filename}c.jpeg',
99                             augmented)
100    fp1[6 * i + 3] = augmented
101
102    brightness = 1 + random.choice([-1, 1]) * 0.1
103                * random.random()
104    augmented = datagen.apply_transform(x=img_array * 255.,
105                                       transform_parameters={
106                                           'brightness': brightness})
107    Image.fromarray(augmented.astype('uint8'))
108        .convert("RGB")
109        .save(f'{save_dir}/{filename}d.jpeg')
110    fp1[6 * i + 4] = augmented
111
112    augmented = rgb_noise(img_array)
113    augmented = np.clip(augmented, 0, 1)
114    matplotlib.image.imsave(f'{save_dir}/{filename}e.jpeg',
115                             augmented)
116    fp1[6 * i + 5] = augmented
117
118    trainy_augm[6 * i] = trainy[i]
119    trainy_augm[6 * i + 1] = trainy[i]
120    trainy_augm[6 * i + 2] = trainy[i]
121    trainy_augm[6 * i + 3] = trainy[i]
122    trainy_augm[6 * i + 4] = trainy[i]
123    trainy_augm[6 * i + 5] = trainy[i]
124    print(f'{i} ')      # For progress display purposes only
125    np.save('trainy_with_augmentation', trainy_augm)
126 #-----
127 # Execute data augmentation
128 #-----
129
130 trainy = np.load('trainy.npy')
131 generation(trainy, df_labels)
```

APÊNDICE B – CÓDIGO PARA MARCAÇÃO DA ÁREA PATOLÓGICA

```

1 def masking(image, explainer, predicted):
2     """
3     Create a mask of the most important regions of an image.
4
5     This function takes an image represented as a numpy array, an
6     explainer object and an integer indicating the predicted
7     class of the image. The function uses the explainer object to
8     calculate the SHAP values of the image and then creates a mask
9     of the most important regions that contributed to the model's
10    prediction. The function then applies some image processing
11    techniques such as thresholding, dilation and closing to the
12    mask to remove noise and improve the visibility of the
13    important regions. The function returns both the raw mask and
14    the original image with the mask applied.
15
16    Parameters:
17    image (numpy array): The image to be masked.
18    explainer (shap.Explainer): An explainer object that has been
19    already trained on the image data.
20    predicted (int): The predicted class of the image.
21
22    Returns:
23    tuple: A tuple containing two numpy arrays, the first one is
24    the raw mask and the second one is the original image with the
25    mask applied.
26    """
27    shap_values = explainer(image.reshape((1, 224, 224, 3)))
28    premask = shap_values.values[:, :, :, :, predicted]
29    premask = np.abs(premask)
30    premask = np.sum(premask.reshape(224,224,3), axis=2) / 3
31    premask /= np.max(premask)
32    premask = np.clip(premask*255, 0, 255)
33    _, th = cv2.threshold(premask.astype('uint8'), 50, 255,
34                          cv2.THRESH_BINARY)
35    kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (5,5))
36    eroding = cv2.erode(th, kernel, 6)
37    kernel = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (7,7))
38    opening = cv2.morphologyEx(eroding, cv2.MORPH_OPEN, kernel, 3)
39    dilatation = cv2.dilate(opening, kernel, 3)
40    dilatation = cv2.bitwise_and(dilatation, dilatation,
41                                mask = (np.clip(np.sum(image, axis=2), 0,
42                                                255) > 5).astype('uint8'))
43    masked1 = cv2.bitwise_and(image, image,

```

```
44         mask = dilatation.astype('uint8')
45     not_dilatation = cv2.bitwise_not(dilatation)
46     masked2blur = cv2.bitwise_and(image, image,
47         mask = not_dilatation.astype('uint8'))
48     blurred = cv2.blur(masked2blur, [5,5])
49     blurred = cv2.bitwise_and(blurred, blurred,
50         mask = not_dilatation.astype('uint8'))
51     masked = cv2.bitwise_xor(masked1, blurred)
52     return dilatation, masked
53
54 def output(image, explainer, model):
55     """
56     Create a visual representation of the important regions of an
57     image.
58
59     This function takes an image represented as a numpy array, an
60     explainer object and a trained model. The function first uses
61     the model to predict the class of the image, and then uses the
62     explainer object to calculate the SHAP values of the image.
63     The function then creates a mask of the most important regions
64     that contributed to the model's prediction and applies the mask
65     to the original image. The function plots the original image,
66     the raw mask and the masked image using matplotlib's subplots.
67
68     Parameters:
69     image (numpy array): The image to be visualized.
70     explainer (shap.Explainer): An explainer object that has been
71     already trained on the image data.
72     model: a trained model to make the prediction
73
74     Returns:
75     None
76     """
77     labels = {0: "sem patologia",
78             1: "leve",
79             2: "moderada",
80             3: "severa",
81             4: "proliferativa"
82     }
83     predicted = np.argmax(model.predict(
84         image.reshape((1, 224, 224, 3)), verbose=0))
85     fig, axs = plt.subplots(1, 3, figsize=(24, 8),
86         constrained_layout=True)
87     fig.suptitle("Área relevante da RD\nIntensidade da patologia: "+
88         labels[predicted], size=32)
89     [axi.set_axis_off() for axi in axs.ravel()]
```

```
90     axs[0].set_title("Retinografia do paciente",
91                     fontdict={'fontsize':22},
92                     verticalalignment='center', y=1.05)
93     axs[0].imshow(image)
94     mark, masked = masking(image, explainer, predicted)
95     axs[1].set_title("Área de atenção", fontdict={'fontsize':22},
96                     verticalalignment='center', y=1.05)
97     axs[1].imshow(mark, cmap='gray')
98     axs[2].set_title("Retinografia do paciente com"
99                     +"\ndestaque na área de atenção",
100                     fontdict={'fontsize':22},
101                     verticalalignment='center', y=1.05)
102     axs[2].imshow(masked)
103     plt.show()
104     return mark, masked
```

APÊNDICE C – CÓDIGO PARA TREINAMENTO DA REDE NEURAL

```

1 #(...)
2 def best_model():
3     """
4     Create a neural network model based on the best values from
5     optimizer.
6
7     This function returns a neural network model based Keras Tuner
8     optimal values.
9
10    Parameters:
11    None
12
13    Returns:
14    model: a training model to make the prediction
15    """
16    input_shape = (224, 224, 3)
17    img_input = Input(shape=input_shape, dtype=tf.uint8)
18    x = tf.cast(img_input, tf.float32)
19    x = densenet_preprocess_input(x)
20    base_model = DenseNet121(include_top=False,
21                            pooling="max",
22                            weights='imagenet')
23    base_model.trainable = True
24    x = base_model(x)
25    x3 = Flatten()(x)
26    x3 = Dense(units=1056, activation="tanh")(x3)
27    x3 = Dense(units=352, activation="tanh")(x3)
28    x3 = Dense(units=96, activation="relu")(x3)
29    x3 = BatchNormalization()(x3)
30    x3 = Dropout(0.25)(x3)
31    predictions = Dense(5, activation='softmax')(x3)
32    model = Model(inputs=img_input,
33                 outputs=predictions)
34    model.compile(optimizer=tf.keras.optimizers.Adam(0.00027),
35                 loss='categorical_crossentropy',
36                 metrics=[tf.keras.metrics.CategoricalAccuracy(
37                         name="cat_acc"),
38                         tf.keras.metrics.AUC(name='auc'),
39                         tf.keras.metrics.Recall(name='recall'),
40                         tf.keras.metrics.Precision(name='precision')])
41    return model
42
43
44

```



```
91         initial_epoch=hist.epoch[-1],
92         callbacks=[es, mc, csvlogFine])
93     model = tf.keras.models.load_model('bestmodel.h5')
94     preds = model.predict(valx, verbose=1)
95     evaluation_model = model.evaluate(x=valx, y=valy, verbose=1)
96     print(evaluation_model)
97     save_csv(evaluation_model)
98
99 #-----
100 # Carregar rótulos
101 #-----
102
103 trainy = np.load('trainy_with_augmentation.npy')
104 valy = np.load('valy.npy')
105 id_trainy_cl0 = np.where(trainy==0)[0]
106 id_trainy_cl1 = np.where(trainy==1)[0]
107 id_trainy_cl2 = np.where(trainy==2)[0]
108 id_trainy_cl3 = np.where(trainy==3)[0]
109 id_trainy_cl4 = np.where(trainy==4)[0]
110 trainy = np.array(to_categorical(trainy), dtype='uint8')
111 valy = np.array(to_categorical(valy), dtype='uint8')
112
113 #-----
114 # Carregar imagens
115 #-----
116
117 trainx = np.memmap('trainx_DA_uint8.dat',
118                   dtype='uint8',
119                   mode='r',
120                   shape=(len(trainy), 224, 224, 3))
121 valx = np.memmap('valx_uint8.dat',
122                 dtype='uint8',
123                 mode='r',
124                 shape=(len(valy), 224, 224, 3))
125 #-----
126 # Downsampling
127 #-----
128
129 id_cl0_downsample = resample(id_trainy_cl0,
130                              replace=True,
131                              n_samples=len(id_trainy_cl4),
132                              random_state=42)
133 id_cl1_downsample = resample(id_trainy_cl1,
134                              replace=True,
135                              n_samples=len(id_trainy_cl4),
136                              random_state=42)
```

```
137 id_cl2_downsample = resample(id_trainy_cl2,
138                               replace=True,
139                               n_samples=len(id_trainy_cl4),
140                               random_state=42)
141 id_cl3_downsample = resample(id_trainy_cl3,
142                               replace=True,
143                               n_samples=len(id_trainy_cl4),
144                               random_state=42)
145 idx = id_cl0_downsample
146 idx = np.append(idx, id_cl1_downsample)
147 idx = np.append(idx, id_cl2_downsample)
148 idx = np.append(idx, id_cl3_downsample)
149 idx = np.append(idx, id_trainy_cl4)
150 trainx = np.array([trainx[i] for i in idx])
151 trainy = np.array([trainy[i] for i in idx])
152
153 #-----
154 # Treinamento e simulação
155 #-----
156
157 model = best_model()
158 running(model, trainx, trainy, valx, valy)
```

APÊNDICE D – SUMÁRIO DA REDE NEURAL TESTADA

```

1 Model: "densenet121_best"
2 -----
3 Layer (type)                Output Shape                Param #
4 =====
5 input_1 (InputLayer)        [(None, 224, 224, 3)]      0
6
7 tf_op_layer_Cast (TensorFlo  (None, 224, 224, 3)      0
8 wOpLayer)
9
10 tf_op_layer_RealDiv (Tensor  (None, 224, 224, 3)      0
11 FlowOpLayer)
12
13 tf_op_layer_BiasAdd (Tensor  (None, 224, 224, 3)      0
14 FlowOpLayer)
15
16 tf_op_layer_RealDiv_1 (Tens  (None, 224, 224, 3)      0
17 orFlowOpLayer)
18
19 densenet121 (Functional)    (None, 1024)              7037504
20
21 flatten (Flatten)          (None, 1024)              0
22
23 dense (Dense)               (None, 1056)              1082400
24
25 dense_1 (Dense)             (None, 352)               372064
26
27 dense_2 (Dense)             (None, 96)                33888
28
29 batch_normalization (BatchN  (None, 96)                384
30 ormalization)
31
32 dropout (Dropout)          (None, 96)                0
33
34 dense_3 (Dense)             (None, 5)                 485
35
36 =====
37 Total params: 8,526,725
38 Trainable params: 8,442,885
39 Non-trainable params: 83,840
40 -----
41 None

```

APÊNDICE E – CÓDIGO PARA MEDIÇÃO DAS MÉTRICAS FINAIS

```
1 #-----
2 # Carregar a rede neural
3 #-----
4
5 best_model = tf.keras.models.load_model('/content/drive/MyDrive/
      bestmodelfinal5.h5')
6
7 #-----
8 # Carregar rótulos
9 #-----
10
11 testy = np.load('/content/drive/MyDrive/testy.npy')
12 testy = np.array(to_categorical(testy), dtype = 'uint8')
13
14 #-----
15 # Carregar imagens
16 #-----
17
18 testx = np.memmap('/content/drive/MyDrive/testx_uint8.dat',
19                  dtype='uint8',
20                  mode='r',
21                  shape=(len(testy), 224, 224, 3))
22
23 #-----
24 # Medição do modelo
25 #-----
26
27 preds = best_model.predict(testx, verbose = 1)
28 evaluation_model = best_model.evaluate(x = testx, y = testy, verbose =
      1)
29 print(evaluation_model)
30 save_csv(evaluation_model)
```

APÊNDICE F – ARTIGO PUBLICADO

1. RIBEIRO, F. R. G.; REGIS, C. D. M. Proposta de Redimensionamento de Imagens de Fundoscopia Utilizada para Classificar a Retinopatia Diabética. In: XII Conferência Nacional em Comunicações, Redes e Segurança da Informação (ENCOM 2022), 2022, Guaramiranga - CE.

ANEXO A – CÓDIGO PARA REDIMENSIONAMENTO DAS IMAGENS (RIBEIRO; REGIS, 2022)

```
1 #(...)
2 file_dir = r'D:\dataset\unzipped'
3 output_dir = r'D:\dataset\resized'
4 if not os.path.exists(output_dir+'\\train'):
5     os.mkdir(output_dir+'\\train')
6 for dirname, _, filenames in os.walk(file_dir):
7     for filename in filenames:
8         img_filename = os.path.join(dirname, filename)
9         img_file = cv2.imread(img_filename)
10        sqr_res = int((np.floor(np.max(img_file.shape)
11                          / dest_res)) * dest_res)
12        if sqr_res > img_file.shape[0]:
13            h_pad = int((sqr_res - img_file.shape[0])/2)
14        else:
15            h_pad = 0
16        if sqr_res > img_file.shape[1]:
17            v_pad = int((sqr_res - img_file.shape[1])/2)
18        else:
19            v_pad = 0
20        img_padded = cv2.copyMakeBorder(img_file, h_pad, h_pad,
21                                       v_pad, v_pad,
22                                       cv2.BORDER_CONSTANT, value=(0, 0, 0))
23        output_filename = os.path.join(output_dir,
24                                       os.path.split(dirname)[-1], filename)
25        output_img = cv2.resize(img_padded,
26                                (dest_res, dest_res),
27                                interpolation=cv2.INTER_NEAREST)
28        cv2.imwrite(output_filename, output_img)
```