

UNIVERSIDADE FEDERAL DO AMAZONAS
FACULDADE DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

MATEUS SARAIVA NASCIMENTO

DETECÇÃO DO PARASITA DA MALÁRIA EM FILMES DE GOTA ESPESSA
UTILIZANDO REDES NEURAS RASAS E TÉCNICAS DE PROCESSAMENTO
DIGITAL DE IMAGENS

MANAUS

2023

UNIVERSIDADE FEDERAL DO AMAZONAS
FACULDADE DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

MATEUS SARAIVA NASCIMENTO

DETECÇÃO DO PARASITA DA MALÁRIA EM FILMES DE GOTA ESPESSA
UTILIZANDO REDES NEURAIAS RASAS E TÉCNICAS DE PROCESSAMENTO
DIGITAL DE IMAGENS

Dissertação apresentada ao Curso de Mestrado em Engenharia Elétrica, área de concentração Controle e Automação de Sistemas e linha de pesquisa Reconhecimento de Padrões e Otimização do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Amazonas.

Orientador: Prof. Dr. Cícero Ferreira Fernandes Costa Filho
Coorientadora: Prof^a. Dr^a. Marly Guimarães Fernandes Costa

MANAUS

2023

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

N244d Nascimento, Mateus Saraiva
Detecção do parasita da malária em filmes de gota espessa utilizando redes neurais rasas e técnicas de processamento digital de imagens / Mateus Saraiva Nascimento . 2023
63 f.: il. color; 31 cm.

Orientador: Cícero Ferreira Fernandes Costa Filho
Coorientadora: Marly Guimarães Fernandes Costa
Dissertação (Mestrado em Engenharia Elétrica) - Universidade Federal do Amazonas.

1. Detecção. 2. Parasita. 3. Malária. 4. Gota espessa. 5. Rede neural rasa. I. Costa Filho, Cícero Ferreira Fernandes. II. Universidade Federal do Amazonas III. Título

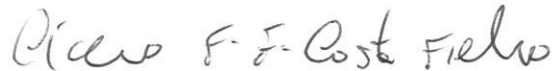
MATEUS SARAIVA NASCIMENTO

**DETECÇÃO DO PARASITA DA MALÁRIA EM FILMES DE GOTA
ESPESSA UTILIZANDO REDES NEURAIS RASAS E TÉCNICAS
DE PROCESSAMENTO DIGITAL DE IMAGENS**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Amazonas, como requisito parcial para obtenção do título de Mestre em Engenharia Elétrica na área de concentração Controle e Automação de Sistemas.

Aprovada em 27 de julho de 2023.

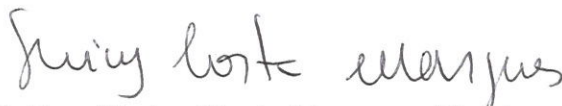
BANCA EXAMINADORA



Prof. Dr. Cícero Ferreira Fernandes Costa Filho

Presidente

Universidade Federal do Amazonas



Prof^a. Dra. Greicy Costa Marques, Membro

Universidade Federal do Amazonas



Prof. Dr. Thiago Brito Bezerra, Membro

Universidade Federal do Amazonas

AGRADECIMENTOS

Agradeço primeiramente a Deus pela minha vida, pela vida da minha família e por ter me dado saúde e conhecimento para superar as dificuldades.

Agradeço aos meus pais e minha irmã, que sempre me incentivaram a estiveram presentes durante a realização deste trabalho de pesquisa.

Agradeço aos meus orientadores, Prof. Dr. Cícero Ferreira Fernandes Costa Filho e Prof^a. Dr^a. Marly Guimarães Fernandes Costa, pela orientação, aprendizado e, principalmente, pela confiança em mim depositada.

Agradeço, finalmente, à Universidade Federal do Amazonas e, em especial, ao Centro de Tecnologia Eletrônica e da Informação – CETELI – por todo trabalho e cuidado para com seus integrantes.

RESUMO

A região amazônica registra cerca de 90% dos casos de malária do Brasil e a transmissão da doença está relacionada diretamente às condições ambientais da região, principalmente no interior da capital, onde o diagnóstico da doença também é tardio devido à escassez de profissionais qualificados (BRASIL, 2020).

Este trabalho traz uma nova perspectiva para a detecção de parasitas da malária em imagens de gota espessa ao utilizar redes neurais simples e técnicas de processamento digital de imagens. O objetivo final deste trabalho é detectar o parasita através de uma rede neural que possa ser embarcada em dispositivos móveis de baixa capacidade de memória, como *smartphones*. Os trabalhos mais recentes utilizam arquiteturas neurais complexas como *Efficient NET* ou *ROENet*, ou redes especializadas para detecção de objetos, tais como *YOLO* e *Faster-RCNN*, o que dificulta embarcar o sistema desenvolvido em dispositivos móveis.

As técnicas apresentadas neste trabalho se baseiam em redes como *perceptron* simples, multicamadas e regressor logístico associados a técnicas como segmentação através do método de *Otsu* e morfologia matemática.

Os melhores resultados obtidos são precisão: 96,33%, sensibilidade: 98,85% e F1-score: 97,57% e superam resultados quando comparados a literaturas publicadas anteriormente. No entanto, não foi possível efetuar uma comparação rigorosa entre todos os trabalhos devido à utilização de diferentes bancos de imagens.

Palavras-chave: Detecção, parasita, malária, gota espessa, rede neural rasa.

ABSTRACT

The Amazon region records about 90% of Brazil's malaria cases and the transmission of this disease is directly related to the region's environmental conditions, mainly within the capital, where the diagnosis is also late due to the shortage of qualified professionals (BRASIL, 2020).

This work brings a new perspective on how to detect malaria parasites in thick blood smear sample images by using neural networks and digital image processing techniques. The ultimate goal of this work is to detect the parasite through a shallow neural network that can be embedded in low-memory mobile devices, such as smartphones. Recent work uses complex architectures such as Efficient NET or ROENet, or specialized object detection networks such as YOLO and Faster-RCNN, making it difficult to board the system developed on mobile devices.

The techniques presented in this work are based on shallow networks such as simple perceptron, multi-layer perceptron and logistic regressor, all of them associated with methods such as segmentation by Otsu and mathematical morphology.

The best results obtained are accuracy: 96.33%, sensitivity: 98.85% and F1-score: 97.57% and outperform previously published literature. However, it is not possible to make a strict comparison between all the work due to the use of different image databases.

Keywords: Detection, parasite, malaria, thick blood smear, shallow neural networks.

LISTA DE FIGURAS

Figura 1 - Punção digital para coleta de sangue para preparo de gota espessa	28
Figura 2 - Elemento estruturante plano	30
Figura 3 - Elemento estruturante não-plano	30
Figura 4 - Exemplos de ES	31
Figura 5 - Exemplos de erosão	32
Figura 6 - Ilustração de um neurônio biológico	34
Figura 7 - Ilustração de um neurônio artificial	35
Figura 8 - Rede de propagação direta com uma única camada de neurônios.....	37
Figura 9 - Rede de propagação direta com uma camada oculta e uma camada de saída	38
Figura 10 - Rede recorrente sem neurônios ocultos	39
Figura 11 - Divisão de dados do conjunto de imagens.....	42
Figura 12 - Aquisição de dados de malária baseada em smartphone	43
Figura 13 - Diferentes tamanhos de parasitas.....	43
Figura 14 - Avaliação estatística do conjunto de dados	44
Figura 15 - Diagrama de blocos da etapa de pré-processamento	45
Figura 16 - Remoção de ruídos através de fechamento morfológico	46
Figura 17 - Processo de binarização da imagem de entrada pelo método de Otsu.....	46
Figura 18 - Processo de exclusão de WBCs.....	47
Figura 19 - Imagem de entrada após aplicação da máscara contendo possíveis parasitas	47
Figura 20 - Remoção de ruídos e pequenos componentes.....	49
Figura 21 - Arquiteturas das redes neurais utilizadas.....	50
Figura 22 - Desempenho do MSE durante o treinamento do <i>perceptron</i> para detecção de parasitas.....	54
Figura 23 - Desempenho do MSE durante o treinamento do regressor logístico para detecção de parasitas	55
Figura 24 - Relação entre valores preditos corretamente e incorretamente durante o treinamento do regressor logístico.....	55
Figura 25 - Desempenho do MSE durante o treinamento do <i>perceptron</i> multicamadas para detecção de parasitas	56
Figura 26 - Exemplificação do problema	56

LISTA DE TABELAS

Tabela 1 – Parâmetros do treinamento e características da arquitetura <i>perceptron</i> simples	50
Tabela 2 – Parâmetros do treinamento e características da arquitetura do regressor logístico.	50
Tabela 3 – Parâmetros do treinamento e características da arquitetura <i>multilayer perceptron</i> .	50
Tabela 4 – Desempenho da arquitetura Perceptron simples com três tamanhos de filtros para remoção de WBCs.....	55
Tabela 5 – Desempenho das arquiteturas	56
Tabela 6 – Matriz de confusão para arquitetura <i>perceptron</i> simples	56
Tabela 7 – Matriz de confusão para arquitetura do regressor logístico.....	56
Tabela 8 – Matriz de confusão para arquitetura <i>perceptron</i> multicamadas	56
Tabela 9 – Comparação dos resultados obtidos com a arquitetura 1 com outros resultados obtidos na literatura	57

LISTA DE QUADROS

Quadro 1 - Resumo da análise realizada nos trabalhos relacionados	24
---	----

LISTA DE SIGLAS

ACWE	<i>Active Contour Without Edges</i>
AUC	<i>Area Under the Curve</i>
CNN	<i>Convolutional Neural Network</i>
ELM	<i>Extreme Learning Machine</i>
ES	Elemento Estruturante
FN	Falso Negativo
FP	Falso Positivo
FPN	<i>Feature Pyramid Network</i>
IGMS	<i>Iterative Global Minimum Screening</i>
kNN	<i>k-Nearest Neighbor</i>
MLP	<i>Multilayer Perceptron</i>
MS	Milissegundos
MSE	<i>Mean Squared Error</i>
OMS	Organização Mundial de Saúde
ReLU	Unidade Linear Retificadora
RNA	Rede Neural Artificial
RNN	<i>Randomized Neural Networks</i>
ROC	<i>Receiver Operating Characteristic</i>
ROI	<i>Region of Interest</i>
RVFL	<i>Random Vector Functional Link</i>
SNN	<i>Schmidt Neural Network</i>
SVM	<i>Support Vector Machine</i>
VN	Verdadeiro Negativo
VP	Verdadeiro Positivo
WBC	<i>White Blood Cell</i>
WHO	<i>World Health Organization</i>

SUMÁRIO

1	INTRODUÇÃO.....	13
1.1	Objetivo Geral.....	15
1.2	Objetivos Específicos.....	15
1.3	Organização do Trabalho.....	15
2	REVISÃO DA LITERATURA.....	17
2.1	Análise dos trabalhos do grupo 1.....	17
2.2	Análise dos trabalhos do grupo 2.....	19
2.3	Análise dos trabalhos do grupo 3.....	20
3	REFERENCIAL TEÓRICO.....	27
3.1	Diagnóstico laboratorial da Malária.....	27
3.1.1	O preparo da gota espessa.....	27
3.2	Morfologia Matemática.....	28
3.2.1	Erosão.....	30
3.2.2	Dilatação.....	31
3.2.3	Abertura e Fechamento.....	31
3.3	Redes Neurais.....	32
3.3.1	<i>Perceptron</i>	32
3.3.2	Treinamento de redes neurais.....	35
3.3.3	Arquitetura de redes neurais.....	36
4	MATERIAIS E MÉTODOS.....	40
4.1	Materiais.....	41
4.2	Métodos.....	43
4.2.1	Aplicação de Técnicas de PDI para Extração de ROIs.....	44
4.2.2	Extração de Características dos Pixels das ROIs.....	47
4.2.3	Filtragem de ruídos.....	47
4.2.4	Arquiteturas das Redes Neurais Utilizadas.....	48

4.2.5	Parâmetros do treinamento das redes	49
4.2.6	Métricas de avaliação	51
4.2.7	Definição do ambiente de trabalho	52
5	RESULTADOS	53
5.1	Apresentação dos resultados do <i>perceptron</i> simples	53
5.2	Apresentação dos resultados de treinamento do regressor logístico.....	53
5.3	Apresentação dos resultados de treinamento do <i>perceptron multicamadas</i>	55
5.4	Apresentação dos resultados de teste	55
5.5	Problemas e soluções	57
6	DISCUSSÃO	58
7	CONCLUSÕES E TRABALHO FUTUROS.....	59
8	REFERÊNCIAS	60

1 INTRODUÇÃO

De acordo com o documento “Bases do Diagnóstico Microscópico da Malária. Parte 1: Guia do Aluno” (BRASIL, 2020), em todo o mundo, a malária continua a ser um dos mais relevantes problemas de saúde pública e, em nosso país, ela também persiste como uma das principais questões, em especial na região Amazônica, que registra cerca de 90% dos casos e onde a transmissão da doença está diretamente relacionada às condições ambientais e socioculturais. Mas é na região extra-amazônica que a malária apresenta maior letalidade, seja devido ao diagnóstico tardio, seja por manejo clínico inadequado dos casos esporádicos importados de áreas endêmicas ou mesmo autóctones em poucos estados.

A malária, mundialmente um dos mais sérios problemas de saúde pública, é uma doença infecciosa causada por protozoários do gênero *Plasmodium* e transmitida ao ser humano por fêmeas de mosquitos do gênero *Anopheles*, produzindo febre e outros sintomas. Quatro espécies de plasmódio podem causar a doença: *P. falciparum*, *P. vivax*, *P. malariae* e *P. ovale*.

Alguns fatos sobre a malária, de acordo com Bases do Diagnóstico Microscópico da Malária Parte II (BRASIL, 2020):

- Em todo o mundo, estima-se haver 247.000.000 (247 milhões) de casos de malária por ano.
- Aproximadamente 85% de todos os casos ocorrem na África ao sul do Saara.
- Estima-se que ocorram 881.000 mortes por malária todo ano, das quais 91% ocorrem na África, principalmente entre crianças menores de 5 anos.
- Uma criança morre de malária a cada 30 segundos.
- O *Plasmodium falciparum* é a mais perigosa das quatro espécies de parasitos que causam malária em humanos.
- Em termos financeiros, calcula-se que a malária custe mais de US\$ 12 bilhões de dólares no mundo todo.

Tradicionalmente, o diagnóstico da doença é feito pela visualização microscópica do plasmódio em exame da gota espessa de sangue. Apesar de a microscopia ser considerada o padrão-ouro para o diagnóstico e monitoramento do tratamento da malária, essa técnica exige pessoal treinado e experiente no exame de distensões sanguíneas.

A Organização Mundial de Saúde – OMS (World Health Organization – WHO, 2021) mostra que, apesar de nos países de baixa renda, a incidência e número de mortes no mundo

devido à malária tenham diminuído entre 2000 e 2019, a malária ainda é a 6ª causa de morte mais frequente. Também representa um alto risco para viajantes e migrantes. Por isso, a OMS recomenda que o diagnóstico precoce e o tratamento rápido sejam os primeiros alicerces de qualquer programa de controle.

Novas técnicas científicas estão sendo empregadas para desenvolver diagnósticos simples, eficazes e passíveis de realização fora do laboratório. Sabendo-se que a chave para a redução da taxa de mortalidade é o diagnóstico precoce e uma terapia eficaz, espera-se que a detecção automática do parasita da malária possibilite diagnósticos rápidos nas comunidades locais, assegurando o tratamento imediato e adequado para prevenir a disseminação da doença.

Desde 2015 pesquisas são realizadas com o objetivo de encontrar uma arquitetura simples e eficaz para a detecção do parasita. Em Aris et al. (2020), as imagens do tipo gota espessa foram segmentadas e classificadas usando agrupamento rápido de *K-Means*, usando espaço de cores RGB, CMY, XYZ, HSV e YUV. Os componentes de cores são extraídos e agrupados em três categorias, que são WBCs e parasitas da malária, manchas e fundo. Usando um conjunto de dados de 100 imagens com 1294×980 pixels, os autores relataram uma precisão de 99,74%, uma sensibilidade de 77,30% e uma especificidade de 99,76%.

Delahunt et al. (2015) propõem uma combinação de uma máquina de vetores de suporte (*support vector machine* - SVM) linear e uma rede neural convolucional para classificação após localizar área de candidatos a parasitas. Eles sugerem o uso de redes neurais convolutivas (*convolutional neural networks* - CNNs) para extração de recursos e seus resultados relatados em 143 pacientes (usando apenas características tradicionais como características morfológicas, de cor, textura e outros). Os resultados obtidos conseguiram prever um limite de detecção de cerca de 300 parasitas/ μL e apresentaram uma especificidade de 92%.

Mehanian et al. (2017) primeiro detecta as células brancas sanguíneas (*white blood cell* - WBC) aplicando uma SVM em possíveis áreas de interesse e depois treinam outra SVM em candidatos de parasitas, que são gerados usando um método de limiar local dinâmico. Finalmente, usam uma CNN para extração e classificação dos objetos detectados. Os autores relatam que o método alcançou uma sensibilidade de 91,6%, precisão de 89,7% e especificidade de 94,1%. O conjunto de dados foi constituído de 1452 imagens de 195 pacientes. No entanto, o tempo de execução é de aproximadamente 20 minutos. Para a detecção de parasitas, esse método proposto não é necessariamente mais rápido que a detecção humana.

Os exemplos acima demonstram redes complexas e que há uma busca regular por um método simples, rápido e eficaz para a detecção da malária, porém como se pode ver no último

exemplo, nem todos os autores atingem o objetivo esperado, seja na eficácia ou no tempo de processamento da rede, o que dificulta embarcar os sistemas desenvolvidos em dispositivos de baixa capacidade de memória.

1.1 Objetivo Geral

Propor e avaliar um método para a detecção do parasita da malária, fundamentado na associação de redes neurais e processamento digital de imagens.

1.2 Objetivos Específicos

- Avaliar o desempenho de um *perceptron* simples na detecção de parasitas da malária;
- Avaliar o desempenho do regressor logístico e do *perceptron* multicamadas na detecção de parasitas da malária;
- Utilizar o processamento digital de imagens para acelerar o processo de detecção do parasita da malária com classificadores neurais

1.3 Organização do Trabalho

Este trabalho está organizado conforme a divisão descrita a seguir:

- Capítulo 1: Introdução;
- Capítulo 2: Revisão da Literatura;
- Capítulo 3: Referencial Teórico;
- Capítulo 4: Materiais e Métodos;
- Capítulo 5: Resultados e Discussões;
- Capítulo 6: Trabalhos Futuros.

O capítulo 1 apresenta o contexto do trabalho, discorrendo sobre a incidência da doença da malária, a importância do controle dessa doença e quais os objetivos gerais e específicos deste trabalho.

O capítulo 2 apresenta trabalhos que foram publicados na literatura na área de detecção do parasita da malária em imagens de gota espessa ou filme fino por meio de técnicas de aprendizado de máquina, redes neurais convolucionais e aprendizado profundo.

O capítulo 3 apresenta os fundamentos teóricos das técnicas utilizadas no desenvolvimento deste trabalho: morfologia matemática, redes neurais e suas arquiteturas, bem como o desenvolvimento na segmentação de imagens e os métodos de otimização avaliados.

No capítulo 4 são apresentados os materiais e métodos utilizados nessa proposta: características do banco de dados e detalhes da metodologia proposta.

No capítulo 5 são apresentados os resultados alcançados e efetua-se uma discussão dos mesmos.

No capítulo 6 são apresentadas sugestões de trabalhos futuros com recomendações de melhoria em relação ao método proposto.

2 REVISÃO DA LITERATURA

Para verificar quais as técnicas, no estado da arte, adotadas na detecção de parasitas da malária em imagens microscópicas de filme espesso, foi necessário realizar uma pesquisa bibliográfica sistemática a respeito do tema. Para essa busca os seguintes termos foram utilizados:

Diagnosis OR Disease OR detection OR classification (Título) AND Malaria (Título) AND thick* blood (Tópico) AND parasite (Todos os campos) AND smear (Todos os campos) AND image (Todos os campos) and 2022 or 2021 or 2020 or 2019 or 2018 (Anos da publicação).

As pesquisas foram realizadas nas bases literárias IEEE Xplore, Web of Science e Engineering Village. Os artigos foram obtidos utilizando a mesma estratégia de busca avançada com as devidas conversões e adaptações. Foi feito também a busca de artigos que utilizaram o mesmo banco de imagens. Buscou-se principalmente por artigos de revista, por apresentarem maior credibilidade e confiabilidade, e deu-se preferência pelos mais recentes.

Após a análise desses trabalhos, verificou-se que a aparência das imagens microscópicas de filme espesso, utilizadas para detecção do parasita da malária, pode apresentar diferentes propriedades de cor, dependendo da iluminação utilizada. Dessa forma, torna-se bastante desafiador a detecção desses parasitas utilizando-se ferramentas automáticas.

Na pesquisa bibliográfica constatou-se que 3 grupos de técnicas distintas são utilizadas para a identificação dos parasitas. Os artigos do grupo 1 utilizam técnicas de processamento digital de imagem, e filtros para remoção de ruídos na imagem. Nos artigos do Grupo 2, as seguintes operações são empregadas para detectar parasitas da malária: segmentação → extração característica → classificador (*support vector machines*, *perceptron* e *k-means*). No grupo 3, as técnicas de aprendizado profundo são usadas para classificação de *patches*, classificação da imagem ou para detecção de parasitas em imagens completas. Uma abordagem para detecção de parasitas em imagens completas envolve a obtenção de *patches* e a classificação dos mesmos através de redes de aprendizagem profunda.

A seguir será feita uma análise dos trabalhos pertencentes a esses três grupos.

2.1 Análise dos trabalhos do grupo 1

2.1.1 *Segmentation of malaria parasite candidates from thick blood smear microphotographs image using active contour without edge* (S. R. Abidin, U. Salamah, and A. S. Nugroho, 2016)

Nesta pesquisa, os autores propuseram um método de segmentação do parasita da malária compreendendo as seguintes etapas: aprimoramento da imagem através de um filtro passa-baixa para redução do ruído e a segmentação do parasita usando a Técnica de Contorno Ativo sem Borda (ACWE). Na sequência, os autores aplicaram filtros morfológicos para remover conteúdos indesejados da imagem (erosão, dilatação e melhoria de contraste) e, finalmente, aplicaram uma operação de limiar. Em um conjunto de dados de 50 imagens com 140×140 pixels, não especificada pelos autores, foi relatada uma precisão de 97,57% e uma taxa de FP igual a 12,04%. Finalmente, pode-se observar que ainda há espaço para melhoria neste trabalho, visto que alguns parasitas não foram muito bem segmentados devido à intensidade do contraste ser muito baixa e as transições de cores entre objeto e fundo não estarem bem definidas em algumas imagens. Como trabalho futuro, os autores sugeriram o estudo da redução do custo computacional do ACWE e redução da taxa de FP.

2.1.2 Computer aided diagnosis of Malaria disease for thin and thick blood smear microscopic images (I. R. Dave and K. P. Upla, 2017)

Os autores usaram campos microscópicos com 3136×2352 pixels, onde cada campo microscópico é captado em 10 planos de foco diferentes. Os parasitas reais são obtidos através de uma anotação manual das coordenadas x-y nas imagens. A origem do banco de imagens não é citada. No pré-processamento, realiza-se a filtragem de ruído utilizando-se os valores máximos de filtro Laplaciano obtidos de 10 planos de foco diferentes. Para a detecção dos parasitas, os autores primeiro separam os objetos do primeiro plano (parasitas, glóbulos brancos, plaquetas e ruído) do fundo, através do limiar adaptativo aplicado no histograma do plano de saturação (coordenada S do sistema de cores HSV). Depois, os objetos que não correspondem a parasitas do campo (glóbulos brancos, plaquetas e ruídos) são eliminados usando o recurso de área dos componentes conectados. Segundo os autores, a área máxima de parasitas é de 700 pixels. Usando esse limite de área, outros objetos que não os parasitas são eliminados. Os autores relataram uma precisão de 90,81% e uma taxa de falso positivos de 10,12%.

A novidade do algoritmo reside na sua robustez e eficácia tanto para esfregaços finos quanto para gota espessa. Além disso, pode-se notar que o uso de aprendizado não supervisionado para classificação torna o algoritmo mais rápido e menos complexo do que as outras técnicas de aprendizado supervisionado existentes.

2.2 Análise dos trabalhos do grupo 2

2.2.1 *Image analysis for malaria parasite detection from microscopic images of thick blood smear (I. R. Dave, 2017)*

Nesta pesquisa, o autor sugere a seguinte sequência de etapas para detecção do parasita da malária: redução de variações da imagem devido à vários fatores, como iluminação e contraste; segmentação usando a técnica de limiar de histograma adaptativo do canal de saturação do espaço HSV seguido de operações morfológicas para redução de ruídos na imagem; extração de características, como formato, textura, momentos centrais, momentos invariantes de Hu., momentos de Zernike e área, dentre outros. Essas características são utilizadas como entrada de uma máquina vetorial de suporte. Foi utilizada uma base de dados com 87 imagens de campos microscópicos, com 3136×2352 pixels. A origem do banco de imagens não foi mencionada. O autor relatou uma sensibilidade de 86,34% e uma especificidade de 96,60%.

É importante ressaltar que a eficácia desse algoritmo pode melhorar caso sejam utilizadas mais imagens para treinamento do classificador e este estudo pode ser estendido para detecção de espécies distintas de *plasmodium* e seus diferentes estágios de vida.

2.2.2 *Fast k-means clustering algorithm for malaria detection in thick blood smear (Aris et al., 2020)*

O objetivo desta pesquisa foi investigar o desempenho da segmentação para melhorar a detecção de malária em imagens de esfregaço de sangue espesso por meio do algoritmo de agrupamento rápido de k-means em vários modelos de cores.

Primeiro, 100 imagens de esfregaço de sangue espesso de tamanho 1294×980 foram segmentadas e classificadas usando agrupamento rápido k-means (FKM). Foram utilizados 15 componentes de cores dos seguintes espaços: RGB, CMY, XYZ, HSV e YUV com o objetivo de descobrir qual componente de cor é o melhor para a detecção de parasitas da malária. Os componentes de cores extraídos foram divididos em três categorias: WBCs e parasitas (preto), manchas e ruídos (cinza) e plano de fundo (branco). Foi feita a remoção de ruídos através do descarte de objetos conectados contendo menos de 7 pixels e mais de 200 pixels. Os parasitas conectados foram separados através do algoritmo *watershed*.

Finalmente, os resultados deste trabalho revelam que o melhor desempenho de segmentação é a segmentação através do componente R de RGB com 99,81% de precisão.

2.3 Análise dos trabalhos do grupo 3

2.3.1 *Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images (Rajamaran et al., 2018)*

Neste estudo, a base de dados utilizada é constituída por imagens de lâminas de esfregaço de filme fino de 150 pacientes infectados e 50 pacientes normais. O *dataset* consiste em 27558 imagens entre infectados e não-infectados.

Os modelos de predição foram analisados através do método de validação cruzada de 5 pastas, deixando assim cada pasta com aproximadamente 5500 imagens por pasta, onde metade são imagens parasitadas e a outra metade não-infectada.

Os modelos foram treinados no sistema Windows com Intel Xeon CPU E5-2640v3, processador de 2.60-GHz, 1 TB HDD, 16 GB RAM, Nvidia GTX 1080 Ti, Matlab R2017b, Python 3.6.3, Keras 2.1.1 com Tensorflow 1.4.0 e CUDA 8.0/cuDNN 5.1.

A CNN proposta possui 3 camadas convolucionais e 2 camadas totalmente conectadas. A entrada é composta da imagem redimensionada para o tamanho 100x100. As camadas convolucionais utilizam filtros de tamanho 3x3 com *stride* igual a 2 *pixels*. A primeira e a segunda camada convolucional possuem 32 filtros e a terceira possui 64.

Foi feita a avaliação da detecção do parasita nas células com base na anotação manual de células infectadas e não infectadas. Para isso, foi aplicado um esquema de correspondência ponto a ponto: para cada célula segmentada, foi verificado o número de pontos *ground-truth* na região segmentada. Se houvesse exatamente um ponto na região, isso seria considerado como um verdadeiro positivo (TP). Se não houvesse nenhum ponto na região, isso seria tido como um falso positivo (FP). Se houvesse mais de um ponto na região, falso negativo (FN). Estas contagens permitiram então calcular os valores apresentados para valor preditivo positivo (PPV), sensibilidade e *F1-score*. Foi obtido um PPV de 94,4%, sensibilidade de 96,2% e *F1-score* de 95,2%.

2.3.2 *An ensemble-based approach for automated medical diagnosis of malaria using EfficientNet (G. Marques, A. Ferreras e I. de la Torre-Diez, 2022)*

Nesta pesquisa, os autores propuseram um método para dar suporte à decisão na detecção do parasita da malária através de redes neurais convolucionais. A arquitetura proposta é baseada em *EfficientNetB0* e usa modelos pré-treinados para transferência de aprendizado. Os resultados foram obtidos utilizando-se validação cruzada com 10 pastas. Foram utilizadas 33.070 imagens, provenientes de dois bancos de imagens: *USA National Institutes of Health* e *Chittagong Medical College Hospital*. A técnica proposta seleciona diferentes conjuntos de dados de

amostras para treinamento e teste. Para reduzir a polarização, cada uma das imagens é usada 9 vezes no processo de treinamento, e apenas 1 vez na fase de teste. Os resultados atingidos foram: acurácia de 98,28%, *recall* de 98,82%, precisão de 97,74%, *F1-score* de 98,28% e AUC de 99,76%. Conforme os autores, esses resultados são ótimos, principalmente quando se leva em conta que o banco de imagens para treinamento da rede proposta é diferente do banco de dados utilizado para validação. Essa última afirmativa dos autores pareceu-nos desnecessária, pois em aprendizado de máquina não se calcula o erro de um classificador através da utilização do mesmo conjunto para treinamento e teste.

Finalmente, pode-se concluir que a rede neural *EfficientNet* é confiável quando se trata da detecção do parasita da malária, porém versões mais complexas da mesma rede podem melhorar ainda mais o desempenho da rede, exigindo assim bancos de dados maiores.

2.3.3 Deep Learning for Smartphone-Based Malaria Parasite Detection in Thick Blood Smears (F. Yang et al., 2020)

Como nos trabalhos anteriores, a pesquisa realizada neste trabalho envolveu a detecção do parasita da malária em imagens de sangue com gota espessa. Foi desenvolvido um método que consiste em duas partes: a primeira é a aplicação IGMS (*Iterative Global Minimum Screening*), que executa a triagem rápida da imagem para encontrar áreas com objetos candidatos à parasitas. Assim, na próxima etapa do método foram usados apenas *patches* e não a imagem inteira como entrada. A segunda parte é uma rede neural convolucional customizada, cujo objetivo é a classificação dos objetos encontrados pelo IGMS em parasitas e não parasitas. Foi utilizado um banco de dados do hospital *Chittagong Medical College Hospital*, com 1819 imagens de 150 pacientes diferentes. Após análise do histograma em escala de cinza das imagens, é verificado que os núcleos dos WBCs e dos parasitas tem intensidades mais escuras que os pixels do plano de fundo, por isso é possível filtrar apenas parasitas e WBCs através do método de *Otsu*. Depois, os WBCs são separados dos parasitas por terem sua área maior. Através de uma etapa de subtração de imagens, esses últimos foram excluídos, restando apenas objetos candidatas a conter parasitas. Esses objetos foram segmentados em *patches* com área de 22 pixels, e foram utilizados como entrada da rede neural convolucional customizada para treinamento, validação e teste. Foram feitos testes com *patches* de área igual a 22, 36, 44 e 52 pixels, porém o melhor resultado alcançado foi com o de 22 pixels. As melhores métricas obtidas no resultado foram: acurácia de 97,26%, sensibilidade de 82,73%, especificidade de 98,39%, precisão de 78,98%, F-score de 80.81% e AUC de 97,34%. A grande contribuição

deste trabalho foi o desenvolvimento do algoritmo para detecção através de *smartphones* e o tempo de execução foi de 10 segundos.

2.3.4 Malaria parasite detection in thick blood smear microscopic images using modified YOLOV3 and YOLOV4 models (F. Abdurahman, K. A. Fante, and M. Aliy., 2021)

Os autores afirmaram que os modelos de redes CNN YOLOv3 e YOLOv4 não são otimizados para detectar pequenos objetos, como parasitas da malária em imagens microscópicas, portanto modificaram esses modelos, aumentando a escala de recursos e adicionando mais camadas de detecção para melhorar sua capacidade de detectar pequenos objetos sem diminuir a velocidade de detecção. Nesses trabalhos, os autores usaram um banco de dados de 1182 imagens RGB, com 750×750 pixels, capturados com um *smartphone* anexado ao microscópio disponibilizados por J. Quinn et al. (2014). Os melhores resultados obtidos pela rede CNN YOLOv4 foram os seguintes: Precisão média: 96,32%, recall: 94%, F1-Score: 94%, IOU médio: 62,12%.

A maior parte da contribuição deste trabalho foi a adaptação das arquiteturas YOLOV3 e YOLOV4 para a detecção de objetos pequenos, como são os parasitas da malária. Salienta-se também que a qualidade das imagens obtidas através de um *smartphone* é inferior às aquelas obtidas por câmeras específicas para obtenção de imagens microscópicas.

2.3.5 Deep Learning Approach for Malaria Parasite Detection in Thick Blood Smear Images (H. A. Nugroho and R. Nurfauzi, 2021)

Nesse estudo, os autores propuseram o reconhecimento de parasitas da malária através do aprendizado profundo, mais precisamente utilizando a rede profunda *Faster-RCNN*, com adição de outras duas arquiteturas, FPN (*Feature Pyramid Network*) e ResNet50. O intuito da adição dessas duas arquiteturas foi reduzir a porcentagem de falsos positivos encontrados em trabalhos previamente mencionados pelos autores. É explicado então que a *Faster-RCNN* tem três processos principais: primeiro, uma rede neural convolucional padrão, cujo objetivo é extrair características da imagem de entrada. Em segundo lugar, a FPN é responsável por selecionar locais candidatos a conter o parasita da malária, que foram posteriormente chamados de região de interesse (*region of interest* – ROI). Em terceiro lugar, a classificação da saída do FPN resulta em duas classes: objeto de interesse e área de fundo. Os autores utilizaram dois bancos de imagens retirados do sítio <http://air.ug/downloads/plasmodium-phonecamera.zip>. Um deles foi utilizado para treinamento com 948 imagens e o outro para teste do algoritmo proposto com

38 imagens. Os dados do primeiro banco de imagens foram divididos randomicamente em 80% para treinamento, 10% para validação e 10% para teste. Para o treinamento foi utilizado método de aumento de dados, com o objetivo de evitar o *overfitting* do algoritmo proposto. Os resultados alcançados foram: mAP de 72%, precisão de 78,4% e falso positivo de 3,55% por imagem, uma ótima redução em comparação com trabalhos anteriores, que apresentavam índice de falso positivo igual a 9,28% por imagem.

Verifica-se que alguns fatores afetaram negativamente o desempenho obtido pelo método proposto no trabalho, como os diferentes tipos de parasitas apresentados nos bancos de dados e as diferente colorações e resoluções das imagens.

No Quadro 1 apresenta-se um resumo dos trabalhos dos grupos 1, 2 e 3 revisados anteriormente. Para cada trabalho, as seguintes características são citadas: banco de imagens, método para detecção dos parasitas da malária e resultados.

Quadro 1 – Resumo da análise realizada nos trabalhos relacionados.

Grupo	Artigo	Autores	Conjunto de dados	Técnica	Métricas/Desempenho
1	<i>Segmentation of malaria parasite candidates from thick blood smear microphotographs image using active contour without edge</i>	S. R. Abidin, U. Salamah, and A. S. Nugroho, 2016	Banco não especificado 150 imagens de gota espessa	Técnica de Contorno Ativo sem Borda (ACWE) Morfologia (erosão e dilatação) Limiar de Otsu	Precisão: 97,57% Falso Positivo: 12%
	<i>Computer aided diagnosis of Malaria disease for thin and thick blood smear microscopic images</i>	I. R. Dave and K. P. Upla, 2017	Banco não especificado 30 imagens – esfregaço delgado 87 imagens – gota espessa	Limiar adaptativo – HSV Componentes conectados	Precisão: 90,81% Falso Positivo: 10,8%
2	<i>Image analysis for malaria parasite detection from microscopic images of thick blood smear</i>	I. R. Dave, 2017	Banco não especificado 87 imagens de gota espessa	Limiar adaptativo – HSV Momentos centrais, invariantes de Hu e Zernike Máquina vetorial de suporte	Sensibilidade: 86,34% Especificidade: 96,60%

<i>Grupo</i>	<i>Artigo</i>	<i>Autores</i>	<i>Conjunto de dados</i>	<i>Técnica</i>	<i>Métricas/Desempenho</i>
	<i>Fast k-means clustering algorithm for malaria detection in thick blood smear</i>	Aris et al., 2020	<i>Department of Microbiology & Parasitology, Hospital Universiti Sains Malaysia (HUSM)</i> 100 imagens de gota espessa	Extração de cores em diferentes planos Fast k-means Segmentação watershed	Acurácia: 99,81% Sensibilidade: 55,24% Especificidade: 99,85%
3	<i>Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images</i>	Rajamaran et al., 2018	<i>Chittagong Medical College Hospital</i> 27558 imagens de filme fino	CNN Customizada + regressão logística multinomial usando <i>stochastic gradient descent</i> (SGD)	Acurácia: 98,6% Sensibilidade: 98,1% Especificidade: 99,2% AUC: 99,9% F1-score: 98,7%
	<i>An ensemble-based approach for automated medical diagnosis of malaria using EfficientNet</i>	G. Marques, A. Ferreras e I. de la Torre-Diez, 2022	<i>USA National Institutes of Health (NIH) e Chittagong Medical College Hospital</i> 33070 imagens de gota espessa	CNN EfficientNetB0 + aprendizado por transferência	Acurácia:98,28% Recall: 98,82% Precisão:97,74% F1-score: 98,28% AUC: 99,76%

<i>Grupo</i>	<i>Artigo</i>	<i>Autores</i>	<i>Conjunto de dados</i>	<i>Técnica</i>	<i>Métricas/Desempenho</i>
	<i>Deep Learning for Smartphone-Based Malaria Parasite Detection in Thick Blood Smears</i>	F. Yang et al., 2020	<i>Chittagong Medical College Hospital</i> 27558 imagens de gota espessa	IGMS + CNN Customizada	Acurácia: 97,26% Sensibilidade: 82,73% Especificidade: 98,39% Precisão: 78,98% F1-score: 80.81% AUC: 97,34%
	<i>Malaria parasite detection in thick blood smear microscopic images using modified YOLOV3 and YOLOV4 models</i>	F. Abdurahman, K. A. Fante, and M. Aliy., 2021	J. Quinn et al. (2014) 1182 imagens de gota espessa	YOLOV3 e YOLOV4	Precisão média: 96,32% recall: 94% F1-Score: 94% IOU médio: 62,12%
	<i>Deep Learning Approach for Malaria Parasite Detection in Thick Blood Smear Images</i>	H. A. Nugroho and R. Nurfauzi, 2021.	http://air.ug/downloads/plasmodium-phonecamera.zip 986 imagens de gota espessa	Faster-RCNN + FPN e ResNet50	mAP = 0,72 Precisão: 78,4% Falso Positivo: 3,55%

3 REFERENCIAL TEÓRICO

3.1 Diagnóstico laboratorial da Malária

Em todo o mundo, a malária continua a ser um dos mais relevantes problemas de saúde pública. Em nosso país também persiste como uma das principais doenças, em especial na região Amazônica, que registra cerca de 90% dos casos e onde a transmissão da doença está diretamente relacionada às condições ambientais e socioculturais. No entanto, é fora da Amazônia que a malária apresenta maior letalidade, seja devido ao diagnóstico tardio, seja por manejo clínico inadequado dos casos esporádicos importados de áreas endêmicas ou mesmo autóctones em poucos estados.

A pesquisa do plasmódio exige alto grau de clareza e nitidez para o reconhecimento dos pequenos parasitos da malária numa gota espessa e pode ser feita tanto através do exame de gota espessa de sangue como através do exame de esfregaço delgado (Manual de Diagnóstico Laboratorial da Malária; 2009).

Este trabalho foi realizado com lâminas de gota espessa, pois nelas o parasita tem cor arroxeada ou vermelho escuro, destacando-se assim do fundo da imagem, que tem cor mais clara. Portanto, esse trabalho aborda métodos de segmentação utilizando espaço de cores RGB e HSV, bem como morfologia matemática, técnicas consideradas de leve processamento computacional e, por último, o algoritmo do *Perceptron*, cujo objetivo é aprender e classificar corretamente os pixels da imagem que contém parasitas.

3.1.1 O preparo da gota espessa

A melhor preparação para o diagnóstico de malária é obtida com amostra de sangue colhida diretamente por punção digital ou venosa sem anticoagulante, como mostrado na Figura 1. Após a coleta, a lâmina deve ser mantida em temperatura ambiente para secagem da gota de sangue – pode-se também utilizar estufa de 37°C ou lâmpada de 25-40 watts sob a placa de vidro.

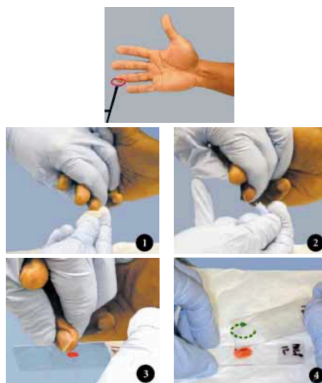


Figura 1: Punção digital para coleta de sangue para preparo de gota espessa

Fonte: Manual de Diagnóstico Laboratorial da Malária (2009)

Vantagens:

- Por concentrar maior quantidade de sangue desmembrado numa área relativamente pequena, a gota espessa aumenta a probabilidade de se encontrar parasitas, o que a torna o método de eleição para o diagnóstico de malária (e de outros hemoparasitas);
- Por ser desmembrada, o processo de coloração é mais rápido, permitindo o processamento de grande número de amostras;
- A distribuição dos parasitas e leucócitos se dá ao acaso em toda a amostra. Portanto, pode-se avaliar a parasitemia contando-se o número de parasitas em relação a um determinado número de leucócitos.

Desvantagens:

- Requer experiência para a identificação de espécies, uma vez que a morfologia do parasita se altera durante o processo de desmembração;
- Requer processamento parcial ou total relativamente rápido depois de colhida a amostra, para evitar a fixação de hemoglobina, a supercoloração e a descoloração.

3.2 Morfologia Matemática

Gonzalez e Woods (2006) definem a linguagem da morfologia matemática como sendo a teoria dos conjuntos. Como tal, a morfologia oferece uma abordagem unificada e poderosa para vários problemas de processamento de imagens. A morfologia matemática foi originalmente desenvolvida para imagens binárias e posteriormente estendida para funções e imagens em tons de cinza. Essa técnica foi originalmente desenvolvida por *Matheron e Serra* na *École des Mines em Paris*, conforme Ian Young (1983). A morfologia pode fornecer algumas informações importantes para o processamento de imagem, são elas: contornos de objetos, esqueletos e segregação de regiões ou conjuntos contendo determinados *pixels*. Por isso, é útil para muitas técnicas de pré e pós-processamento de imagens. Pode também ser usada para construir filtros semelhantes aos filtros espaciais, como suavização, aguçamento ou realce de imagens. Nas operações morfológicas, o processamento da imagem é realizado utilizando-se pequenas máscaras, denominadas de elementos estruturantes (ES).

Normalmente um elemento estruturante é escolhido do mesmo tamanho e forma dos objetos que se deseja processar. Por exemplo, para encontrar linhas em uma imagem, é necessário criar um elemento estruturante linear, para encontrar formas circulares, é necessário criar um ES em forma de disco, e assim sucessivamente. (Dougherty; 1992).

Existem dois tipos de elementos estruturantes: planos e não-planos.

Um elemento estruturante plano, conforme mostra a Figura 2, é uma matriz com valores binários, na qual os *pixels* verdadeiros (1) são incluídos na operação morfológica e os *pixels* falsos (0) não. (Mathworks; 2012).

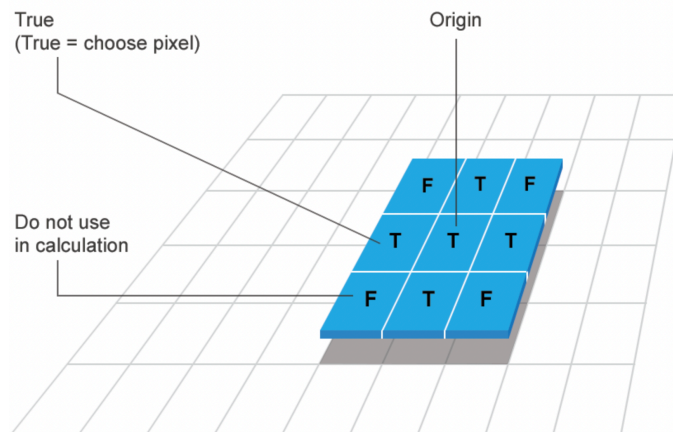


Figura 2: Elemento estruturante plano

Fonte: adaptado de Mathworks (2012)

Um elemento estruturante não-plano, conforme Figura 3, é uma matriz ou “máscara” do tipo *double* e contém valores reais usados na operação morfológica. O *pixel* central da matriz, ou origem, identifica o *pixel* no elemento que está sendo processado. Esses elementos estruturantes não-planos são usados apenas com imagens em tons de cinza. (Mathworks; 2012).

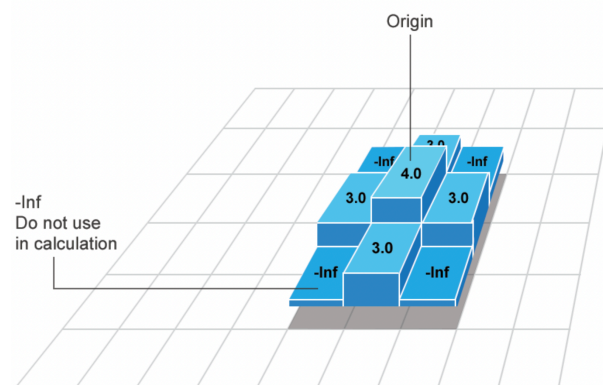


Figura 3: Elemento estruturante não-plano

Fonte: adaptado de Mathworks (2012)

De acordo com Ian Young (1983), o conhecimento sobre um objeto depende da maneira como o observamos. Em particular, a escolha de um determinado elemento estruturante para uma operação morfológica influencia a informação ou característica que se pode obter.

Dougherty (1992) afirma que existem duas propriedades principais que estão diretamente relacionadas aos elementos estruturantes: forma e tamanho.

Forma: o elemento estruturante pode ser uma bola ou uma linha; convexo ou anelar, etc. Ao escolher um determinado elemento estruturante, estabelece-se uma maneira de diferenciar alguns objetos de outros, de acordo com sua forma ou orientação espacial.

Tamanho: um elemento estruturante pode ser uma matriz quadrada 3 X 3 ou 21 X 21. Definir o tamanho do elemento estruturante é definir a escala de observação para diferenciar objetos ou feições da imagem.

Ao trabalhar com imagens, é necessário que os elementos estruturantes sejam arranjos matriciais retangulares. (Gonzalez; Woods; 2006). A Figura 4 apresenta exemplos de ES.

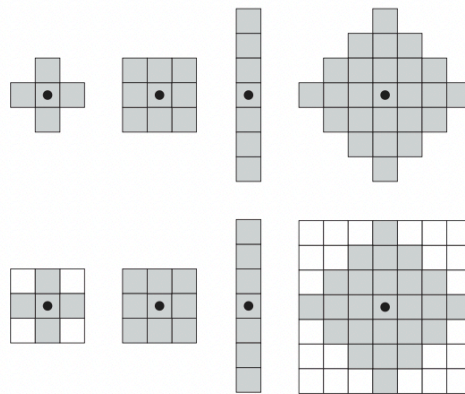


Figura 4: Exemplos de ES

Fonte: Gonzalez e Woods (2006)

Os operadores morfológicos básicos são erosão, dilatação, abertura e fechamento. Essas operações são fundamentais para o processamento morfológico.

3.2.1 Erosão

A equação 1 indica que a erosão de A por B é o conjunto de todos os pontos z de forma que B, transladado por z, está contido em A (Gonzalez; Woods; 2006).

$$A \ominus B = \{z | (B)_z \subseteq A\} \quad (1)$$

Pode-se ver a erosão como uma operação de filtragem morfológica em que os detalhes da imagem menores que o ES são removidos da imagem. Na Figura 5, tem-se a imagem original em 5(a), que irá sofrer a operação de erosão por um elemento estruturante quadrático de tamanho 11x11. Após a operação, obtém-se a figura 5(b). A razão pela qual todas as linhas que

ligam as bordas ao centro da imagem foram removidas, menos as linhas verticais do centro, se deve ao fato de que essas linhas centrais tem largura maior que 11 *pixels*. Ao mudar o elemento estruturante para o tamanho 15x15, obtém-se a figura 5(c) e utilizando um ES de tamanho 45x45, obtém-se a figura 5(d).

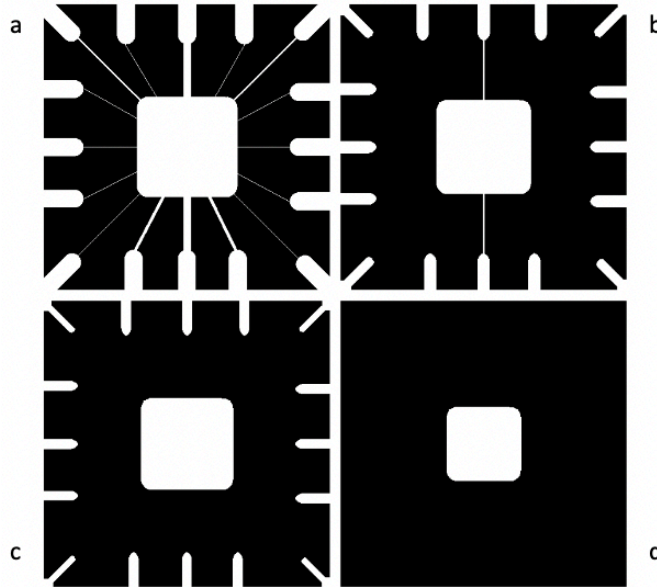


Figura 5: Exemplos de erosão

Fonte: Gonzalez e Woods (2006)

3.2.2 Dilatação

A dilatação de A por B é o conjunto de todos os deslocamentos, z, de forma que \hat{B} e A se sobreponham pelo menos por um elemento, conforme denotado na equação 2. (Gonzalez; Woods; 2006).

$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \emptyset\} \quad (2)$$

Ao contrário da erosão, que é uma operação de diminuição ou afinamento, a dilatação aumenta ou “engrossa” os objetos em uma imagem. A forma específica e a extensão desse espessamento são controladas pelo formato do elemento estruturante utilizado. Uma das aplicações mais simples da dilatação é a eliminação de lacunas em uma imagem.

3.2.3 Abertura e Fechamento

A abertura geralmente suaviza o contorno de um objeto e elimina as saliências finas. O fechamento também tende a suavizar contornos, mas ao contrário da abertura, geralmente funde

as discontinuidades estreitas e elimina pequenos buracos, preenchendo suas lacunas. (Gonzalez; Woods; 2006).

A abertura do conjunto A pelo elemento estruturante B , indicado por $A \circ B$ é definida como:

$$A \circ B = (A \ominus B) \oplus B \quad (3)$$

Assim, a abertura de A por B é a erosão de A por B , seguida de uma dilatação desse resultado por B .

A operação de abertura satisfaz às seguintes propriedades:

- (a) $A \circ B$ é um subconjunto (subimagem) de A .
 - (b) Se C é um subconjunto de D , então $C \circ B$ é um subconjunto de $D \circ B$.
- as $(A \circ B) \circ B = A \circ B$.

O fechamento do conjunto A pelo elemento estruturante B , indicado por $A \bullet B$, é a dilatação de A por B , seguida pela erosão desse resultado por B .

$$A \bullet B = (A \oplus B) \ominus B \quad (4)$$

Da mesma forma, a operação de fechamento satisfaz às seguintes propriedades:

- (a) A é um subconjunto de $A \bullet B$.
- (b) Se C é um subconjunto de D , então $C \bullet B$ é um subconjunto de $D \bullet B$.
- (c) $(A \bullet B) \bullet B = A \bullet B$.

Gonzalez e Woods (2006) indicam uma vantagem imediata da abordagem morfológica sobre o método de filtragem passa-baixa, quando se utiliza operadores morfológicos para unir as lacunas. É nesse caso, a operação morfológica produz diretamente uma imagem binária. A filtragem passa-baixa, por outro lado, começa com uma imagem binária e produz uma imagem em níveis de cinza, o que exigiria uma etapa adicional de limiarização para convertê-la de volta à forma binária.

3.3 Redes Neurais

3.3.1 *Perceptron*

Haykin (2009) afirma que a proposta das redes neurais artificiais foi motivada pelo reconhecimento de que o cérebro humano processa a informação de uma maneira totalmente diferente do computador digital convencional. O cérebro é um computador altamente complexo, não-linear e paralelo. Ele tem a capacidade de organizar seus constituintes estruturais, conhecidos como neurônios, de modo a executar certos cálculos (por exemplo,

reconhecimento de padrões, percepção e controle motor) muitas vezes mais veloz que o computador digital mais rápido existente. Um exemplo disso é quando o cérebro realiza tarefas rotineiras de reconhecimento perceptivo, por exemplo, ao reconhecer um rosto familiar incorporado em uma cena desconhecida, o que ocorre em aproximadamente entre 100 a 200 milissegundos, enquanto que os algoritmos que executam tarefas de muito menor complexidade, em computadores atuais, demoram muito mais tempo .

As redes neurais, à semelhança do cérebro, são constituídas por um grande número de neurônios artificiais. Esses neurônios estão conectados uns aos outros através de uma rede de comunicação complexa, através da qual o cérebro é capaz de realizar procedimentos complexos e tomar decisões (SHALEV-SHWARTasBEN-DAVID, 2014). A arquitetura de um neurônio do sistema nervoso do corpo humano pode ser dividida em três partes: corpo celular, dendritos e o axônio, como pode ser visualizado na Figura 6.

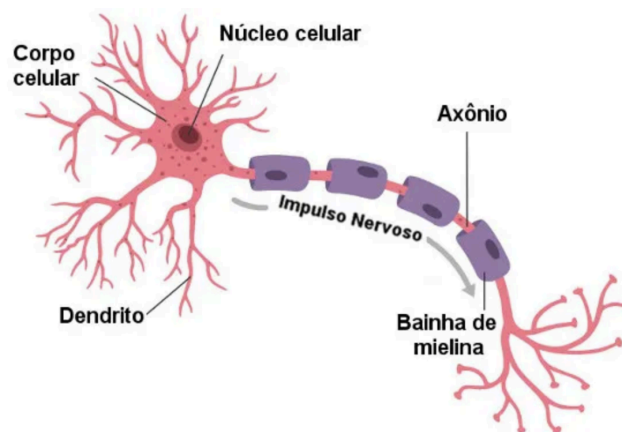


Figura 6: Ilustração de um neurônio biológico

Fonte: Mundo educação (2022)

O neurônio artificial, proposto por McCulloch & Pitts, é mostrado na Figura 7 conforme ilustrado por Barbosa, Dantas e Anjo (2014). As entradas (X) são conectadas ao neurônio através dos pesos (W) que representam os dendritos. O somatório, a função de ativação (θ) e a polarização (b) simbolizam o corpo da célula. A propagação da saída (Y) é idêntica ao axônio num neurônio biológico (AWAD; KHANNA, 2015).

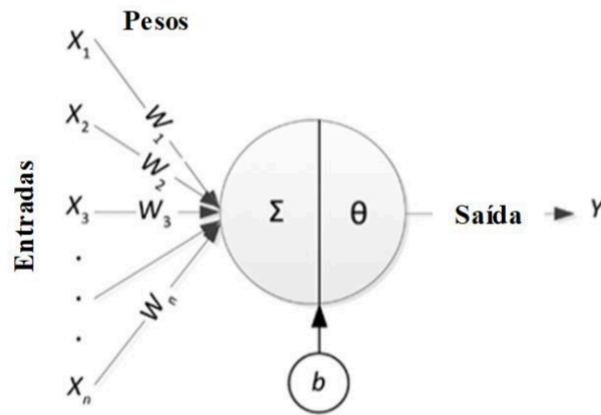


Figura 7: Ilustração de um neurônio artificial

Fonte: Awad e Khanna (2015)

A representação matemática de um neurônio artificial é dada por:

$$Y = \theta \sum_{i=1}^n w_i x_i + b \quad (5)$$

E pode ser modelado usando a forma matricial:

$$Y = \theta(W \cdot X + b) \quad (6)$$

em que,

$$W = [W_1 \ W_2 \ \dots \ W_n] \ e \ X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{bmatrix}. \quad (7)$$

O vetor \mathbf{x} , que representa a entrada, é um conjunto de características apresentado à entrada do neurônio artificial. Essas características influenciam o cálculo da saída y , e diferentes propriedades podem ser obtidas através da manipulação dos componentes do vetor de pesos \mathbf{w} . Para que a rede neural realize uma determinada função, esse vetor de pesos \mathbf{w} é alterada no processo de aprendizagem do algoritmo. A polarização b tem como objetivo permitir um melhor ajuste da função de ativação θ , de acordo com a tarefa a ser desempenhada pela rede neural. A função de ativação prediz o valor da saída e pode ter várias formas, sendo as mais utilizadas:

- Sigmoid: $\theta(a) = \frac{1}{1+e^a}$
- Tangente hiperbólica: $\theta(a) = \frac{e^a - e^{-a}}{e^a + e^{-a}}$
- Degrau: $\theta(a) = \begin{cases} 0 & \text{se } a < 0 \\ 1 & \text{se } a > 0 \end{cases}$

- Linear: $\theta(a) = \begin{cases} 0 & \text{se } a < 0 \\ a & \text{se } 0 \leq a \leq 1 \\ 1 & \text{se } a > 1 \end{cases}$

Uma rede neural é composta por uma camada de neurônios, que por sua vez é composta por vários neurônios artificiais ligados a um conjunto de entrada. A saída de uma camada é um vetor, em que cada item corresponde à saída de um neurônio artificial dessa camada (AWAD; KHANNA, 2015). Na RNA, as camadas adjacentes são inteiramente conectadas entre si, ou seja, todos os neurônios de uma camada anterior estão conectados a todos os neurônios da camada seguinte. Para se obter um desempenho ótimo do modelo a ser treinado, a quantidade de camadas e de neurônios assume um papel relevante. O atrativo principal da solução de problemas através de RNAs é, sem dúvida, a capacidade de aprender através de exemplos e de generalizar a informação aprendida (BRAGA; CARVALHO; LUDERMIR, 2000).

À semelhança de redes neurais, na área de aprendizado de máquina existem diversas ferramentas que realizam as tarefas mais usuais de classificação de padrões. Dentre essas ferramentas, citamos: árvores de decisão, SVM e kNN. Neste trabalho, foram utilizadas as Redes Neurais, mas especificamente o *Perceptron*. Antes de abordar o *Perceptron* em si, no entanto, alguns conceitos de arquitetura e o processo de treinamento de uma rede neural, que são úteis para compreensão, serão abordados.

3.3.2 Treinamento de redes neurais

O processo de treinamento das redes neurais tem por objetivo ensinar uma RNA a reconhecer padrões pertencentes a classes diferentes. O aprendizado consiste no ajuste que é feito nos pesos das camadas das redes, através de um algoritmo de otimização. Observando a equação 8, a saída Y predita pela rede pode não ser igual a saída desejada D , portanto o processo de aprendizado busca reduzir gradualmente o erro entre a saída predita pela rede e a saída desejada, através de um ajuste dos valores dos pesos e polarizações (JAIN; MAO; MOHIUDDIN, 1996).

O erro entre a saída predita e a saída desejada é avaliado por uma função de custo. Através da otimização do valor da função de custo, o algoritmo de otimização ajusta os valores dos pesos e polarizações, para que a saída predita se aproxime ao máximo da saída desejada. O treinamento finaliza quando a RNA alcançar um erro menor que um limiar pré-estabelecido. A função de custo mais utilizada é a função do Erro Médio Quadrático, mostrada na equação 8. Nessa equação, m é o número de amostras do conjunto de treinamento, Y_i a saída predita pela rede e D_i a saída desejada. Podendo ser expressado pela equação 8.

$$E = \frac{1}{m} \sum_{i=1}^m (D_i - Y_i)^2 \quad (8)$$

3.3.3 Arquitetura de redes neurais

A arquitetura de uma rede está associada ao número de camadas da rede e ao número de neurônios em cada camada. Em geral, pode-se identificar três arquiteturas de redes: redes de propagação direta com uma camada única, redes de propagação direta com múltiplas camadas e redes recorrentes (BRAGA; CARVALHO; LUDERMIR, 2000).

3.3.3.1 Redes de propagação direta com camada única

Na forma mais simples de uma rede em camadas, tem-se, conforme mostrado na Figura 8, uma rede com uma camada de entrada e uma camada de neurônios. Em outras palavras, esta rede é uma rede de propagação direta com uma única camada de neurônios. (HAYKIN, 2009). Essa rede é conhecida como *perceptron*.

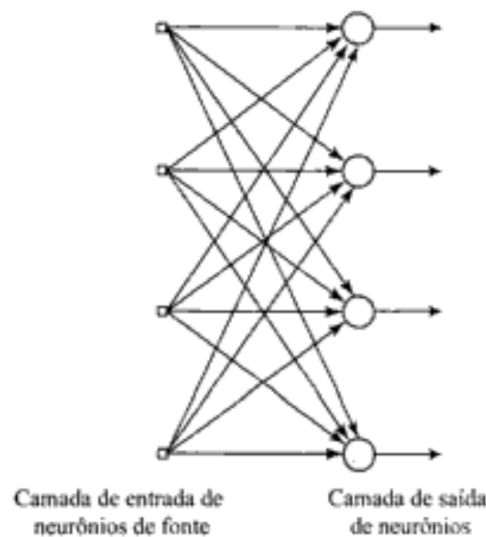


Figura 8: Rede de propagação direta com uma única camada de neurônios

Fonte: adaptado de Haykin (2009)

3.3.3.2 Redes de propagação direta com múltiplas camadas

A segunda classe de redes de propagação direta se distingue pela presença de uma ou mais camadas ocultas de neurônios. Essas redes são conhecidas na literatura por *perceptron* com multicamadas (*multilayer perceptron* – MLP). A função dos neurônios ocultos é introduzir processamento não linear entre a entrada e a saída da rede. Adicionando-se uma ou mais

camadas ocultas, a rede será capaz de extrair características mais complexas, devido ao processamento não linear adicionado (CHURCHLAND; SEJNOWSKI, 1992).

Os nós da camada de entrada da rede fornecem os padrões de ativação da rede neural. Os sinais de saída da primeira camada de neurônios são utilizados como entradas para a segunda camada de neurônios, e assim por diante, conforme mostrado na Figura 9 (HAYKIN, 2009).

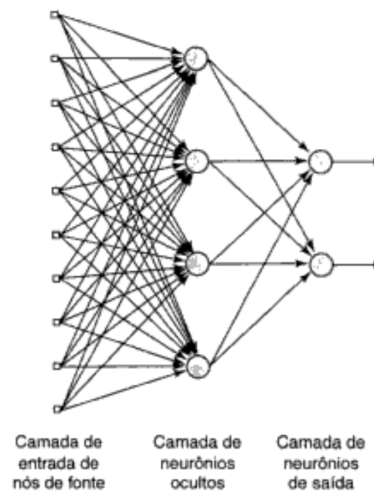


Figura 9: Rede de propagação direta com uma camada oculta e uma camada de saída

Fonte: adaptado de Haykin (2009)

A rede neural da Figura 9 é dita totalmente conectada, pois todos os nós de uma camada da rede estão conectados a todos os nós da camada adjacente seguinte e da camada adjacente anterior.

3.3.3.3 Redes recorrentes

Uma rede neural recorrente se distingue de uma rede neural de propagação direta por ter pelo menos um laço de realimentação. Uma rede recorrente pode consistir, por exemplo, de uma única camada de neurônios com cada neurônio alimentando seu sinal de saída de volta para as entradas de todos os outros neurônios, como ilustrado no grafo arquitetural da Figura 10. Redes cuja saída final é ligada às entradas comportam-se como autômatos reconhecedores de cadeias, onde a saída que é realimentada fornece o estado do autômato (BRAGA; CARVALHO; LUDERMIR, 2000).

A presença de laços de realimentação tem um impacto profundo na capacidade de aprendizagem da rede e no seu desempenho. Além disso, os laços de realimentação envolvem o uso de ramos particulares compostos de elementos de atraso unitário (representados por z^{-1}),

o que resulta em um comportamento dinâmico não-linear, admitindo-se que a rede neural contenha unidades não-lineares (HAYKIN, 2009).

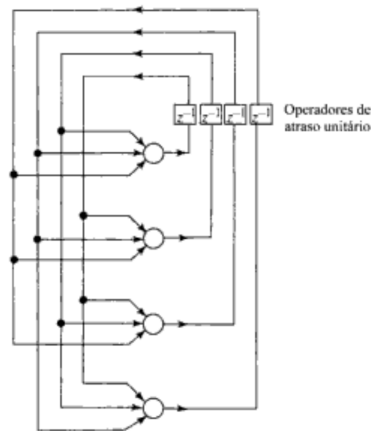


Figura 10: Rede recorrente sem neurônios ocultos

Fonte: adaptado de Haykin (2009)

O perceptron ocupa um lugar especial no desenvolvimento histórico de redes neurais, pois foi a primeira rede neural com um algoritmo de treinamento. A ideia partiu de Frank Rosenblatt em 1958, um psicólogo, que inspirou engenheiros, matemáticos e físicos a continuarem desenvolvendo pesquisas neste ramo (HAYKIN, 2009).

Gurney (1997) afirma que o *perceptron* é a forma mais simples de uma rede neural usada para a classificação de padrões linearmente separáveis. Rosenblatt provou que se os padrões usados para treinar a rede neural forem de duas classes linearmente separáveis, o algoritmo *perceptron* converge e posiciona a superfície da decisão entre as duas classes. Um *perceptron* construído em torno de um único neurônio é limitado a executar a classificação de padrões em apenas duas classes. Ao expandir a camada de saída do *perceptron* para incluir mais de um neurônio, pode-se executar a classificação com mais de duas classes. No entanto, as classes devem ser linearmente separáveis para que o *perceptron* funcione corretamente.

Apenas na década de 1980, é que apareceu o MLP. O treinamento proposto para essa rede com múltiplas camadas foi o algoritmo de retropropagação (*backpropagation*). Nele, quando uma entrada é apresentada à rede, esta faz uma predição da saída e caso haja diferença entre o predito e o desejado, os parâmetros internos da rede sofrem modificações, na tentativa de minimizar o erro.

O *Backpropagation*, se baseia do gradiente descendente, e recebe esse nome pela sua forma de treinamento, que ocorre em sentidos opostos. No primeiro sentido *forward* (para frente), a entrada é apresentada a rede e ela prediz a saída. No segundo *backward* (para trás), se

a saída não for a desejada os pesos das conexões são ajustados e então o processo se repete até que o erro seja minimizado ou outro critério de parada seja atingido.

Uma função de erro muito utilizada é o erro médio quadrático (MSE):

$$E(w) = \sum_{n=1}^N E_n(w) = \sum_{n=1}^N \sum_{k=1}^C (y_k(x_n; w) - t_{nk})^2 \quad (9)$$

Segundo Bishop (2006), de forma resumida, o algoritmo de *backpropagation* é dado seguindo os passos abaixo:

1. Encontrar as ativações de todas as unidades escondidas e de saída aplicando um vetor de entrada x_n a rede;

2. Encontrar o valor de δ_k para todas as unidades utilizando:

$$\delta_k = y_k - t_k \quad (10)$$

J. 3. Obter δ_j de cada unidade escondida através do *backpropagation* da equação:

$$\delta_j = h'(a_j) \sum_k w_{kj} \delta_k \quad (11)$$

4. Obter as derivadas necessárias utilizando:

$$\frac{\partial E_n}{\partial w_{ji}} = \delta_j z_i \quad (12)$$

O algoritmo distribui a contribuição do erro para cada peso na rede, levando em consideração o gradiente local da função de erro (GÉRON, 2017b). Como as funções de ativação de cada neurônio são deriváveis, o algoritmo utiliza-se da regra da cadeia para calcular a derivada do erro com relação a cada parâmetro da rede tendo apenas o valor do erro na última camada. Os métodos de otimização são todos baseados neste algoritmo. Pode-se observar também que o método de treinamento não é tão simples, e devido ao enorme número de pesos na rede ele pode ser demorado.

4 MATERIAIS E MÉTODOS

A metodologia proposta para este trabalho caracteriza-se pela aplicação de morfologia matemática e de redes neurais para realizar a segmentação e classificação de parasitas da malária. A ideia central é utilizar a morfologia matemática em etapas de pré-processamento e segmentação e, posteriormente, realizar a classificação de pixels, como pertencentes ou não a parasitas, através de arquiteturas de RNAs.

No Capítulo 3, foram apresentados os conceitos fundamentais para implementação desse trabalho. Agora, serão apresentados os materiais necessários para a implementação, os métodos aplicados e o detalhamento das arquiteturas dos classificadores propostos.

O banco de dados utilizado neste trabalho contém 1819 imagens e foi dividido em dois conjuntos, um com 910 imagens e outro com 909 imagens. Do conjunto com 910 imagens, foram extraídos pixels de regiões correspondentes a regiões contendo WBCs, parasitas, fundo e ruídos. Esses *pixels* são então utilizados para treinamento de um classificador neural binário. Para o treinamento do classificador neural binário, foram extraídas características de cor de *pixels* das imagens originais, pertencentes às seguintes regiões: WBCs, parasitas e fundo/ruído. A saída do classificador igual a 1 indica um *pixel* pertencente a um parasita, enquanto a saída do classificador igual a 0, indica um *pixel* não pertencente a parasita. Na Figura 11(a) mostramos o processo de treinamento do classificador neural binário.

Seria natural que, após o treinamento desse classificador, o mesmo fosse utilizado para classificar todos os *pixels* de uma imagem, como pertencendo ou não a parasitas. Dessa forma, os parasitas seriam obtidos através da união de *pixels* vizinhos pertencentes a classe parasita. O problema com essa abordagem, é que, como a imagem original contém milhares de pixels, o processo de classificação de todos os pixels seria muito demorado, inviabilizando a utilização do método na prática. Em um experimento que fizemos, para a imagem RGB original, com resolução de 3024×4032 pixels, esse tempo foi de aproximadamente 15 minutos.

Para minimizar o número de *pixels* que serão apresentados ao classificador neural binário, aplicou-se técnicas de processamento digital de imagens (PDI), principalmente operações morfológicas, para selecionar previamente regiões de interesse na imagem (ROIs – *regions of interest*), contendo possíveis parasitas. Apenas os *pixels* pertencentes a essas ROIs são apresentados ao classificador neural. Na Figura 11(b) mostramos como é feito então o processo de detecção de parasitas utilizando as técnicas de PDI e o classificador neural.

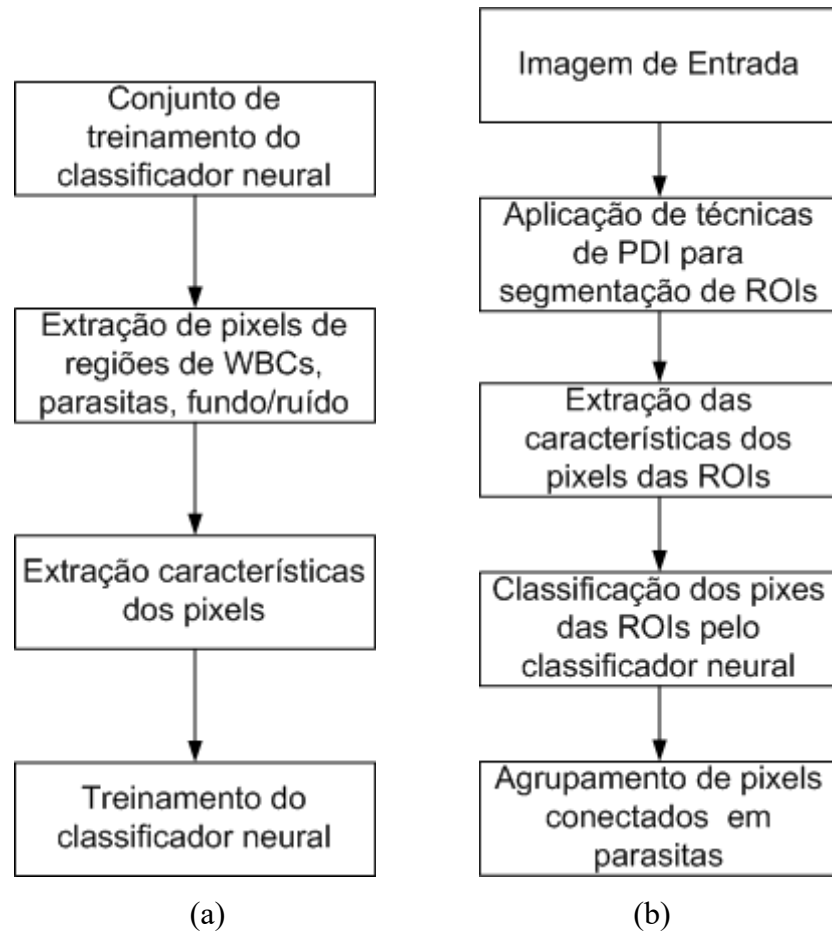


Figura 11: Divisão de dados do conjunto de imagens

4.1 Materiais

A base de dados utilizada nesse trabalho é constituída por imagens de lâminas de esfregaço de sangue espesso de 150 pacientes infectados por *P. falciparum* no *Chittagong Medical College Hospital*, Bangladesh, totalizando 1819 imagens. Para obtenção das imagens, foi utilizada uma câmera de *smartphone* e assim foram obtidas imagens com diversos campos de visão microscópicos. Yang et al. (2020) disponibiliza essas imagens através do link: ftp://lhcfpl.nlm.nih.gov/Open-Access-Datasets/Malaria/Thick_Smears_150. A Figura 12 mostra a configuração do *smartphone*-microscópio e uma captura de tela do telefone exibindo uma imagem de gota espessa. As imagens são capturadas com ampliação de 100x no espaço de cores RGB, com resolução de 3024×4032 pixels.

Um microscopista especializado anotou manualmente cada imagem na *Mahidol-Oxford Tropical Medicine Research Unit (MORU)*, Bangkok, Tailândia, disponibilizando todas as imagens e suas anotações na *National Library of Medicine* (IRB#12972).

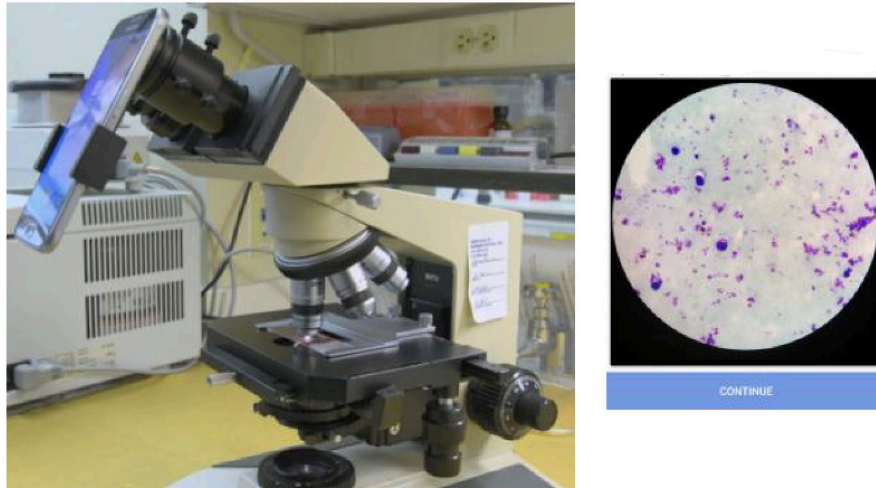


Figura 12: Aquisição de dados de malária baseada em smartphone

Fonte: F. Yang et al. (2020)

No total, 84.961 parasitas foram anotados através de circunferências. O raio dessas circunferências variou entre 2 a 96 *pixels*, com raio médio de 22 *pixels*, conforme mostrado na Figura 13.

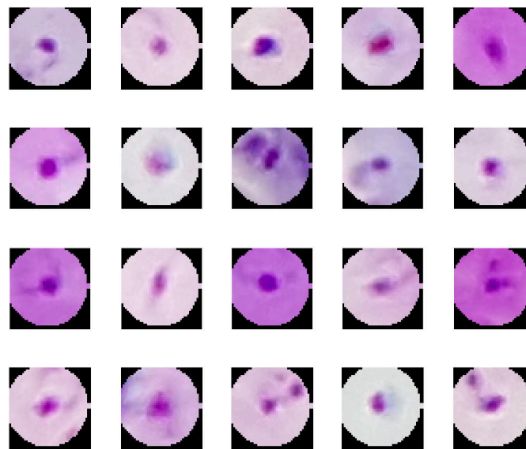


Figura 13: Diferentes tamanhos de parasitas

Fonte: F. Yang et al. (2020)

Cada imagem inclui de 1 a 341 parasitas, com um número médio de 47 parasitas por imagem. Cada paciente contém de 3 a 22 imagens, com um número médio de 12 imagens por paciente. Cada imagem contém de 8 a 3.130 parasitas, com um número médio de 522 parasitas por imagem (F. YANG et al., 2020). A Figura 14 mostra diversos gráficos que ajudam na caracterização da base de dados. Na Figura 14(a) mostra-se um gráfico do raio da circunferência envolvendo o parasita versus o número de parasitas por lâmina. Na Figura 14(b) mostra-se um gráfico do número de parasitas por imagem. Na Figura 14(c) mostra-se um gráfico do número

de imagens por paciente. Na Figura 14(d) mostra-se um gráfico do número de parasitas por paciente.

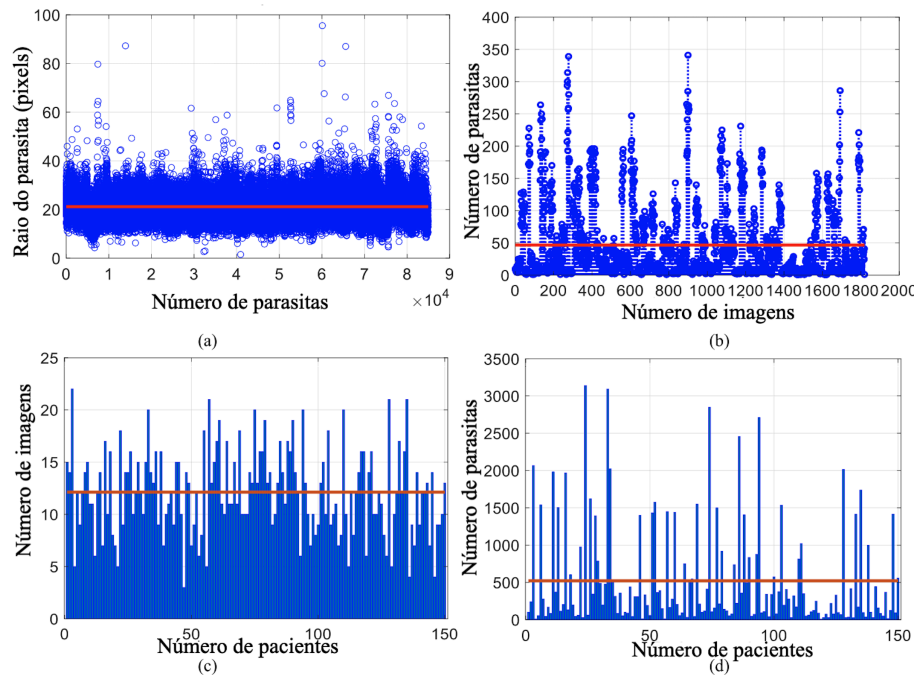


Figura 14: Avaliação estatística do conjunto de dados. (a) raio da circunferência envolvendo o parasita versus o número de parasitas por lâmina; (b) número de parasitas por imagem; (c) número de imagens por paciente; (d) número de parasitas por paciente

Fonte: F. Yang et al. (2020)

Em aprendizado de máquina, uma prática recomendada é utilizar-se diferentes conjuntos de dados para o treinamento, validação e teste do classificador, pois, a utilização de um único conjunto de dados para as três etapas superestima a precisão do algoritmo de aprendizado de máquina.

O conjunto de treinamento é usado para obter os parâmetros dos modelos, que, nesse caso, são os pesos e polarizações dos neurônios das RNAs. Os parâmetros do modelo são ajustados através de um algoritmo de otimização. A etapa de validação é utilizada para a seleção do modelo treinado que resulta em um melhor desempenho no conjunto de validação. Para reportar o resultado do melhor modelo selecionado na etapa de validação, avalia-se o desempenho do mesmo no conjunto de teste. Assim, o conjunto de teste é usado apenas uma vez, no final do processo, para avaliação das métricas de desempenho do melhor modelo.

4.2 Métodos

Na Figura 11(b) mostramos as etapas seguidas nesse trabalho para detecção dos parasitas da malária. Nessa seção, detalharemos cada uma dessas etapas.

4.2.1 Aplicação de Técnicas de PDI para Extração de ROIs

O objetivo desta etapa é obter uma imagem RGB contendo regiões de interesse com possíveis parasitas. Na Figura 15 mostramos os passos utilizados nesta etapa.

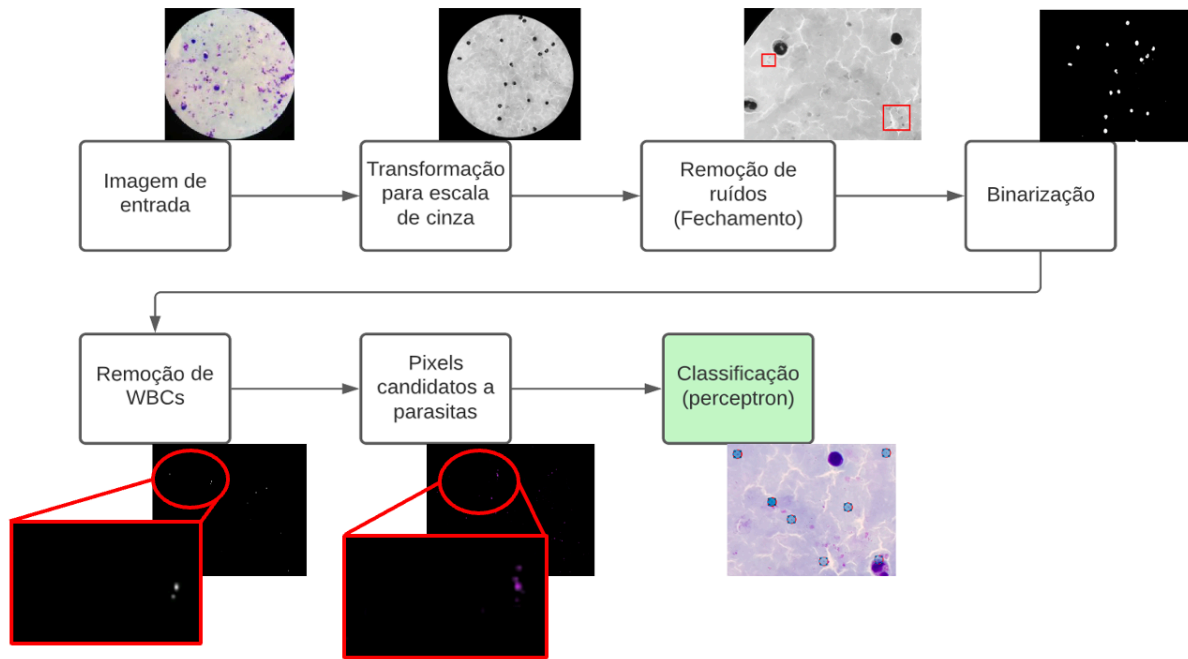


Figura 15: Diagrama de blocos da etapa de pré-processamento

Em primeiro lugar, a imagem a ser processada é carregada em formato RGB a partir do banco de dados. Após o carregamento, a mesma é convertida para níveis de cinza.

Em seguida, utilizamos morfologia matemática com os seguintes objetivos: suavizar o contorno de parasitas e WBCs, unir áreas de pequenos parasitas que, em algumas imagens, encontravam-se bem próximos um dos outros e remoção de ruídos.

O processo utilizado foi o de fechamento morfológico, com elemento estruturante (ES) em forma de disco e com tamanho 4x4. Foi realizado também um teste com elemento estruturante de tamanho 6x6, porém o resultado não foi satisfatório. O resultado dessa etapa de fechamento morfológico é ilustrado na Figura 16. Na Figura 16(a) mostramos uma imagem original em nível de cinza de uma lâmina. A Figura 16(b) mostra uma ampliação de uma região da imagem original contendo parasitas e ruídos. A Figura 16(c) mostra a operação de fechamento morfológico com elemento estruturante 4x4 aplicada na região da Figura 16(b). Observamos que os ruídos foram eliminados. Por outro lado, a Figura 16(d) mostra a operação de fechamento morfológico com elemento estruturante 6x6 aplicada na região da Figura 16(b). Observamos

que não só os ruídos, mas também os parasitas foram removidos, o que gera falsos negativos, prejudicando o desempenho do classificador.

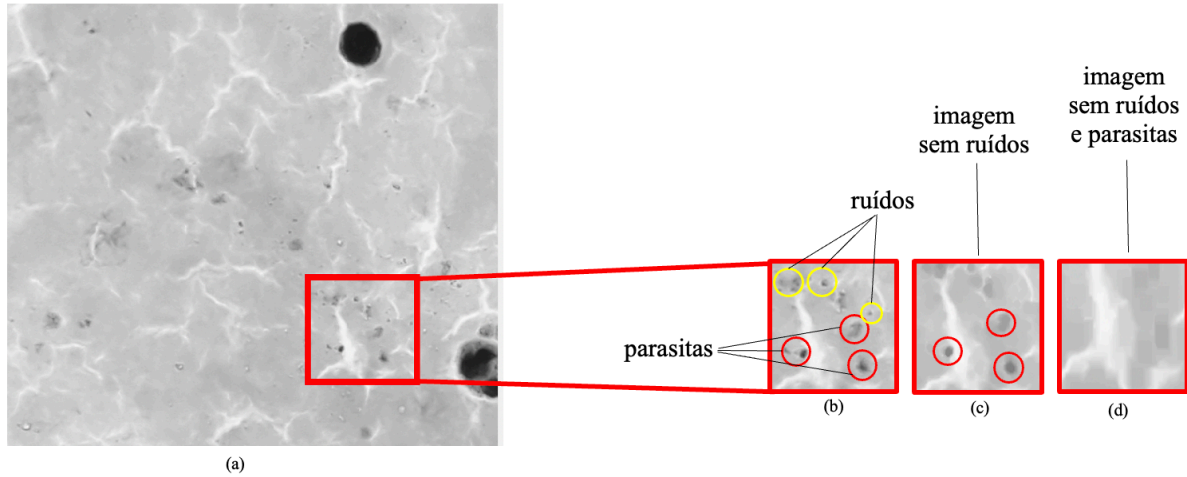


Figura 16: Remoção de ruídos através de fechamento morfológico. (a) imagem original; (b) ampliação de uma região da imagem original contendo parasitas e ruídos; (c) filtragem com elemento estruturante 4x4, mostrando a remoção de ruídos; (d) filtragem com elemento estruturante 6x6 mostrando a remoção de ruídos e parasitas

Após a etapa de fechamento morfológico, ocorre o processo de binarização, que é efetuado através do método de Otsu, um método de limiarização da imagem que usa o equilíbrio do histograma de níveis de cinza da imagem. O algoritmo assume que a imagem se divide em duas classes: o fundo e as regiões de interesse ROIs, e procura encontrar o nível de limiarização ótimo que divide o histograma em duas classes, conforme mostrado na Figura 17. Nesse processo de limiarização, as regiões de interesse são os parasitas e as células WBCs.

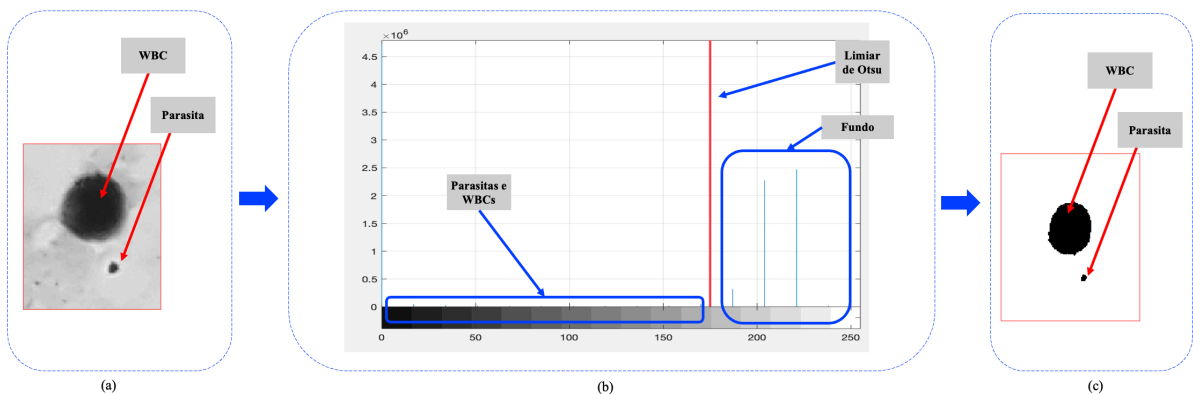


Figura 17: Processo de binarização da imagem de entrada pelo método de Otsu. (a) imagem original; (b) histograma com limiar Otsu; (c) imagem binarizada pelo método de Otsu

Após o processo de binarização, a remoção das células WBCs é realizada através de um filtro de área. Foi feito um estudo sobre qual o melhor tamanho do filtro de área a ser utilizado.

O desempenho do método de detecção do para sita da malária foi avaliado com três tamanhos de filtro, 750, 1000 e 1300, sendo os resultados reportados na Tabela 4 da seção de resultados.

No Matlab, para a implementação desse filtro, foi utilizada a função *bwareaopen*. Conforme será justificado nos resultados, o tamanho ótimo do filtro de área foi definido em 1000 *pixels*. Na Figura 18 mostra-se a extração de uma célula WBC com o filtro de área.

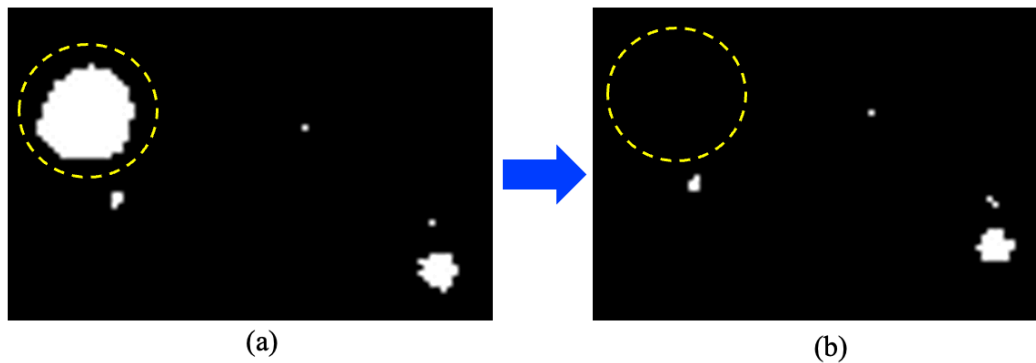


Figura 18: Processo de exclusão de WBCs. (a) imagem original com WBC; (b) imagem após exclusão de WBCs filtro de área

Finalmente, para a seleção dos parasitas na imagem original, utilizou-se a imagem binária resultante das etapas anteriores como máscara. As regiões da imagem original que correspondem ao nível de cinza 1 da imagem binária resultante, corresponde aos parasitas. Essa etapa de marcação na imagem original foi efetuada através da função *bsxfun* do Matlab, a qual pode ser usada para aplicar a operação elementar de multiplicação entre duas matrizes. O resultado final, mostrando apenas os parasitas detectados na imagem original, é mostrado na Figura 19.

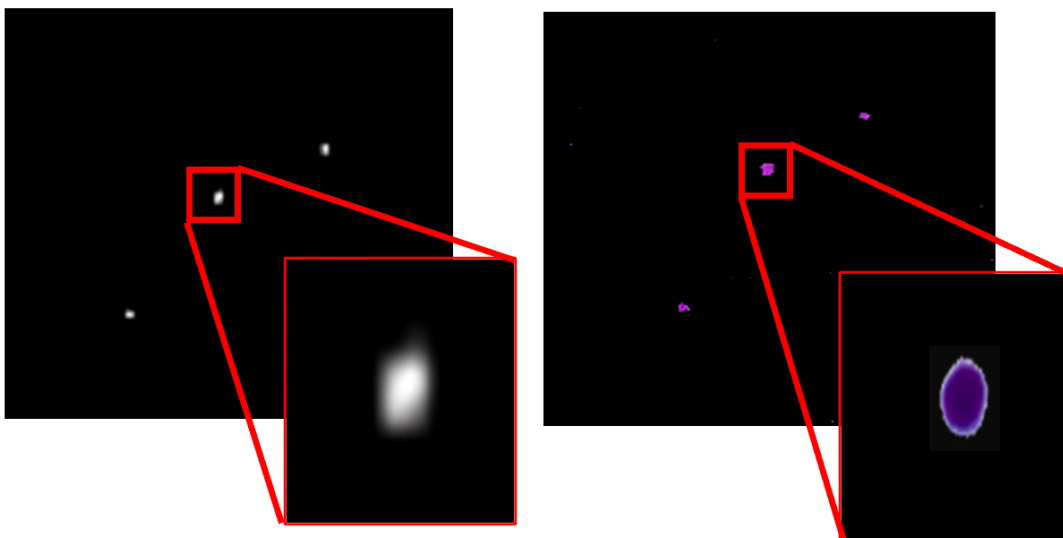


Figura 19: Imagem de entrada após aplicação da máscara contendo possíveis parasitas

Após os candidatos a parasitas serem extraídos, utilizou-se RNAs para a classificação dos *pixels* dos objetos detectados na etapa de pré-processamento em duas classes. *Pixels* pertencentes a parasitas e *pixels* não pertencentes a parasitas. Na saída do classificador tem-se duas possibilidades, 0 ou 1. 0 significa que o *pixel* não pertence a um parasita e 1 significa que o *pixel* pertence a um parasita.

4.2.2 Extração de Características dos Pixels das ROIs

O vetor de características apresentado à entrada dos classificadores contém, para cada *pixel*, as coordenadas do mesmo nos planos de cores RGB e HSV, ou seja, 6 entradas conectadas diretamente à RNA. Esse vetor de características é mostrado através da equação (12):

$$V = [R \ G \ B \ H \ S \ V] \quad (13)$$

4.2.3 Filtragem de ruídos

Nesta seção será mostrada a técnica que removerá ruídos remanescentes, mais precisamente *pixels* que foram erroneamente classificados como verdadeiros (parasitas). Aris et al. (2020) utilizou técnicas de remoção de ruídos que são menores de 7 *pixels* e maiores que 200 *pixels*, objetivando maior acurácia na detecção de parasitas.

Para este trabalho, baseado nos estudos e também após diversas tentativas e erros, foi estabelecido que seriam removidos os objetos classificados como parasitas com área menor que 10 *pixels*.

A remoção dessas áreas é realizada através do mesmo filtro do Matlab que foi utilizado para remoção de WBCs (objetos com área muito grande). O objetivo desta técnica é diminuir o número de falsos positivos (FP), aumentando assim métricas como precisão e F1-score.

A Figura 20(a) mostra, sob anotação vermelha, áreas com menos de 10 *pixels*, enquanto que a única anotação verde é um parasita de área superior ao limiar estabelecido. A Figura 20(b) demonstra a remoção dos ruídos e a conservação da informação do parasita.

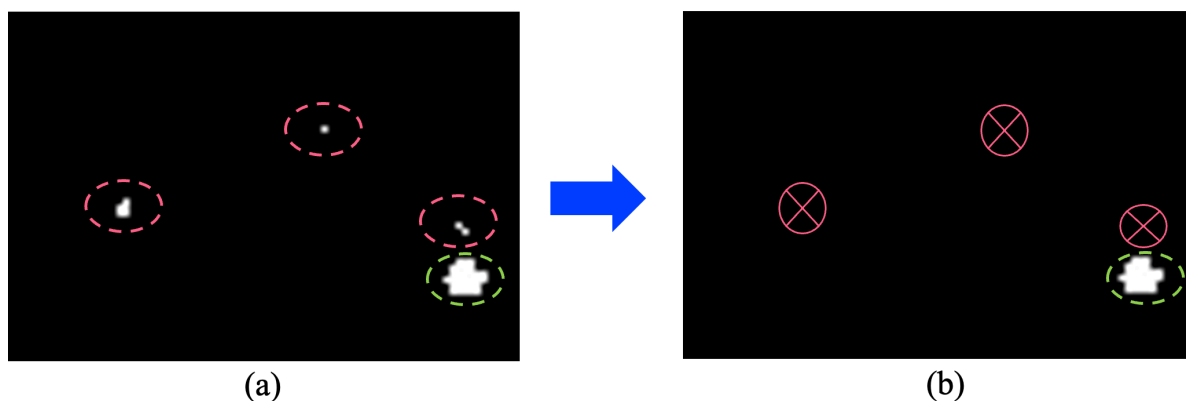


Figura 20: Remoção de ruídos ou pequenos componentes. (a) imagem com ruído (b) imagem sem ruído.

4.2.4 Arquiteturas das Redes Neurais Utilizadas

Nesse trabalho, três classificadores com arquiteturas de redes neurais rasas foram avaliados. A primeira arquitetura foi o perceptron. Sua arquitetura, com apenas um neurônio, é mostrada na Figura 21(a). A função de ativação f é o *hard limit* ou degrau, mostrado na Equação (14). O algoritmo de treinamento da arquitetura 1 é o algoritmo de aprendizado do perceptron, um algoritmo heurístico proposto por Frank Rosenblatt (H AGAN,2020). A segunda arquitetura foi o regressor logístico. É semelhante à arquitetura 1, com a única diferença de que a função de ativação f é a função sigmoide, como mostra a Equação (15). Diferentemente do perceptron, a regressão logística possui uma função de ativação diferenciável, e o treinamento é feito usando o algoritmo de otimização estocástico gradiente descendente. A função de perda do regressor logístico foi a entropia cruzada, que é o logaritmo negativo da probabilidade da classe verdadeira, conforme a equação (16).

$$f(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases} \quad (14)$$

$$f(x) = \frac{1}{1+e^{-x}} \quad (15)$$

$$L_i = -\log(P(c(i)|X_i)) \quad (16)$$

Em que:

L_i – valor da perda quando apresenta – se o padrão X_i na entrada da rede

$c(i)$ é a classe do padrão X_i

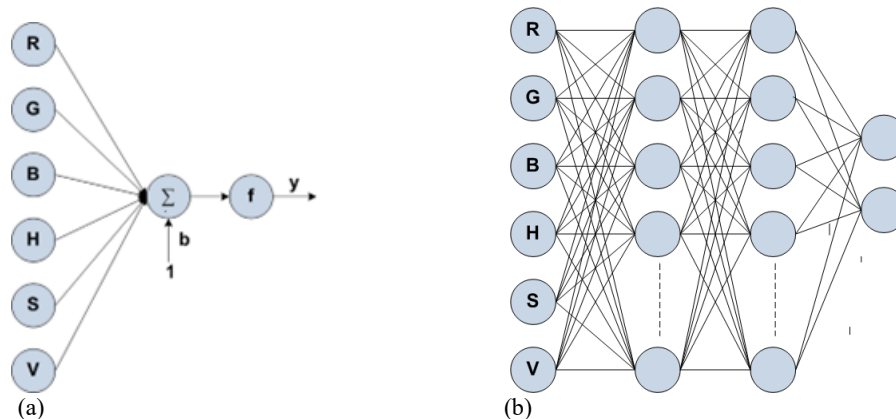


Figura 21: Arquiteturas das redes neurais utilizadas. (a) *perceptron* e regressor logístico. (b) *perceptron* multicamadas.

A terceira arquitetura utilizada foi um perceptron multicamadas, com três camadas de neurônios, conforme a Figura 21(b). A primeira e a segunda camada oculta possuem 15 neurônios (6-15-15-2). A função de ativação dos neurônios da camada oculta é a função ReLU, enquanto a função de ativação dos neurônios da camada de saída é a função *softmax*. A função perda é a mostrada na Equação (16). O algoritmo de retropropagação, com descida de gradiente estocástico, usado no treinamento da terceira arquitetura.

A justificativa para a escolha da arquitetura do *perceptron* multicamadas deveu-se aos seguintes argumentos:

1) Número de camadas: Cybenko (CYBENCO, 1988) mostrou que toda função pode ser aproximada com acurácia arbitrária por uma rede com duas camadas ocultas. Assim, nesse trabalho optou-se por utilizar duas camadas ocultas.

2) Número de neurônios em cada camada: Fletcher (FLETCHER e GOSS, 1993) afirmam que o número de neurônios em cada camada foi determinado baseado no número de entradas da rede. Considerando n o número de entradas, o número mínimo de nós seria de $2n + 1$. O que resultaria em 13 neurônios, esse valor foi arbitrariamente determinado para 15 neurônios por camada oculta.

4.2.5 Parâmetros do treinamento das redes

Para realizar o treinamento das arquiteturas de RNAs é necessário ajustar alguns hiperparâmetros. Na Tabela 1 estão mostrados os valores selecionados para os hiperparâmetros de treinamento utilizados para o *perceptron* simples. Esses valores foram ajustados experimentalmente.

Tabela 1 – Parâmetros do treinamento e características da arquitetura *perceptron* simples.

Parâmetros	Valor
Taxa de aprendizado inicial	0,001
Divisão de dados	70/15/15
Fator de queda de taxa de aprendizagem	0,5
Número máximo de épocas	500
Função de ativação da camada de saída	<i>Hardlim</i>

Na Tabela 2 e 3 estão mostrados os parâmetros referentes ao treinamento da rede utilizando um regressor logístico e um *perceptron* multicamadas, respectivamente.

Tabela 2 – Parâmetros do treinamento e características da arquitetura do regressor logístico.

Parâmetros	Valor
Taxa de aprendizado inicial	0,001
Divisão de dados	70/15/15
Fator de queda de taxa de aprendizagem	0,5
Número máximo de épocas	500
Função de ativação da camada de saída	<i>Sigmoid</i>

Tabela 3 – Parâmetros do treinamento e características da arquitetura *multilayer perceptron*.

Parâmetros	Valor
Taxa de aprendizado inicial	0,001
Divisão de dados	70/15/15
Fator de queda de taxa de aprendizagem	0,5
Número máximo de épocas	500
Neurônios por camada	15
Camadas ocultas	2
Função de ativação da camada de saída	<i>Softmax</i>

O conjunto de dados foi subdividido nos conjuntos de treinamento, validação e teste, com as seguintes proporções: 70% para treinamento, 15% para validação e 15% para teste. Para realizar essa divisão, utilizou-se a função *randperm* do Matlab.

4.2.6 Métricas de avaliação

Para avaliar a qualidade do método proposto para detecção do parasita da malária, foi feita uma comparação com os resultados apresentados em outros trabalhos através das seguintes métricas de desempenho: precisão, sensibilidade e F1-score. Para o cálculo dessas métricas foram utilizados os seguintes valores: *VP* - verdadeiros positivos, que correspondem a parasitas corretamente detectados; *FP* - falsos positivos, que correspondem a ruídos erroneamente classificados como parasitas; *FN* - falsos negativos, que correspondem a parasitas que não foram detectados. Nesse processo de classificação de pixels, não é possível definir o valor *VN* - verdadeiro negativo. Com base nos valores *VP*, *FP* e *FN*, as métricas utilizadas são definidas como:

- **Precisão:** determina a fração de amostras classificadas como positivas que eram realmente positivas.

$$\text{Precisão: } \frac{VP}{VP+FP} \quad (13)$$

- **Sensibilidade:** determina com que frequência o classificador classifica corretamente os exemplos da classe positiva. Quanto maior o valor da sensibilidade, menos *FN* são detectados pelo classificador.

$$\text{Sensibilidade: } \frac{VP}{VP+FN} \quad (14)$$

- **F1-score:** é calculado pela combinação das métricas de precisão e sensibilidade, indica a qualidade geral do modelo e busca estabelecer um equilíbrio entre a precisão e sensibilidade.

$$F1 - score = 2 \times \frac{\text{Precisão} \times \text{Sensibilidade}}{\text{Precisão} + \text{Sensibilidade}} \quad (15)$$

Uma métrica muito utilizadas para avaliar o desempenho de RNA é a área sob a curva ROC (*Receiver Operating Characteristic*), também denominada por AUC (*area under ROC curve*) que representa o poder de um método de classificação. Quanto maior essa área, maior o poder do método. Um método ideal de classificação apresenta AUC=1. Uma classificação randômica apresenta AUC = 0,5 No entanto, a AUC não foi utilizada para avaliar o método proposto, visto que a abordagem escolhida para o desenvolvimento não possibilitou o uso da curva ROC e o cálculo da AUC, pois não é possível determinar o valor *VN*.

4.2.7 Definição do ambiente de trabalho

Para a elaboração deste trabalho, utilizou-se uma conta *Google Colaboratory PRO*, uma ferramenta que permite a alocação e a utilização de *Hardware* na nuvem, acessível através de qualquer navegador. A máquina padrão disponível possui processador Intel® Xeon® CPU @ 2.30GHz *Cache* 56 MB, 16 GB de memória RAM e GPU 16 GB de Memória NVIDIA Tesla P100PCIe.

Para a implementação, utilizou-se a linguagem *Matlab* com as bibliotecas Processamento de Imagem e Visão Computacional e *Machine Learning*.

5 RESULTADOS

Neste capítulo serão apresentados os resultados parciais obtidos referentes à arquitetura 1, que utiliza o *perceptron* simples. Na dissertação serão apresentados os resultados das outras arquiteturas.

5.1 Apresentação dos resultados do *perceptron* simples

A Figura 22 mostra o desempenho do erro quadrático médio durante o treinamento do *perceptron* simples, nos três conjuntos: treinamento, validação e teste.

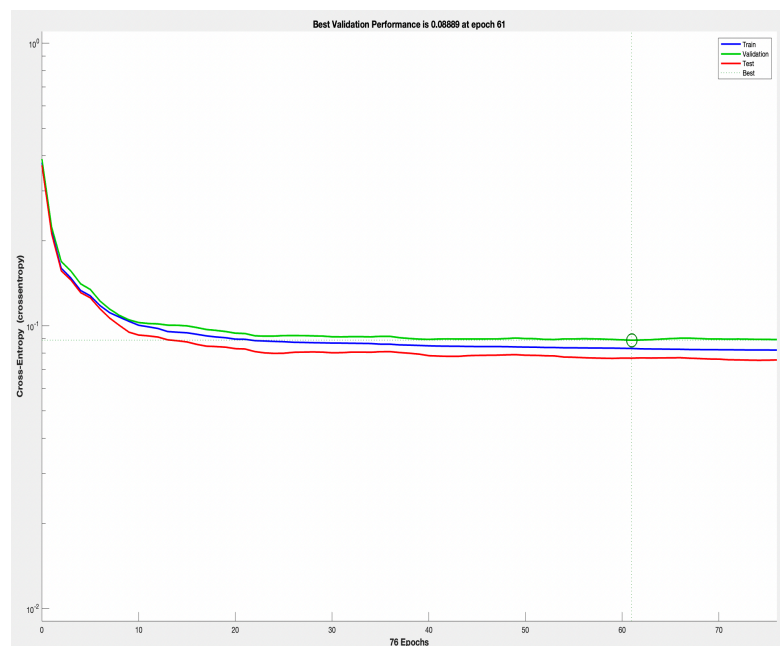


Figura 22: Desempenho do MSE durante o treinamento do *perceptron* para detecção de parasitas

A melhor performance dessa rede é atingida na época 61 e o menor valor do erro quadrático médio (mean square error – MSE) foi de 0,0888.

5.2 Apresentação dos resultados de treinamento do regressor logístico

A Figura 23 mostra o desempenho do erro quadrático médio durante o treinamento do regressor *logístico*, nos três conjuntos: treinamento, validação e teste.

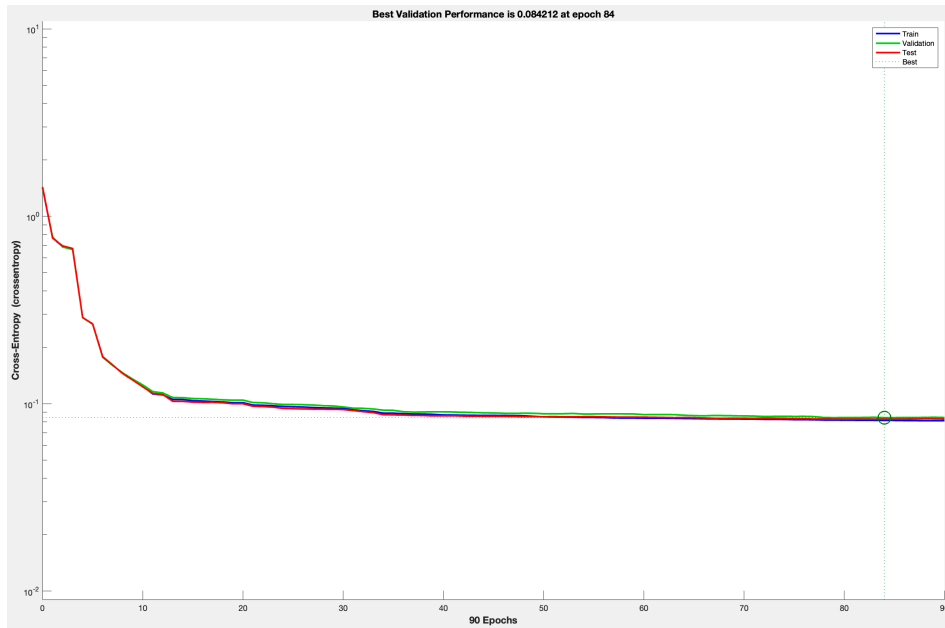


Figura 23: Desempenho do MSE durante o treinamento do regressor logístico para detecção de parasitas

A melhor performance dessa rede é atingida na época 84 e o menor valor de MSE atingido foi de 0,0842. A Figura 24 mostra a relação entre erros e acertos desse modelo num espaço amostral com 400 casos, onde pode-se perceber que não ocorrem altas discrepâncias durante o treinamento.

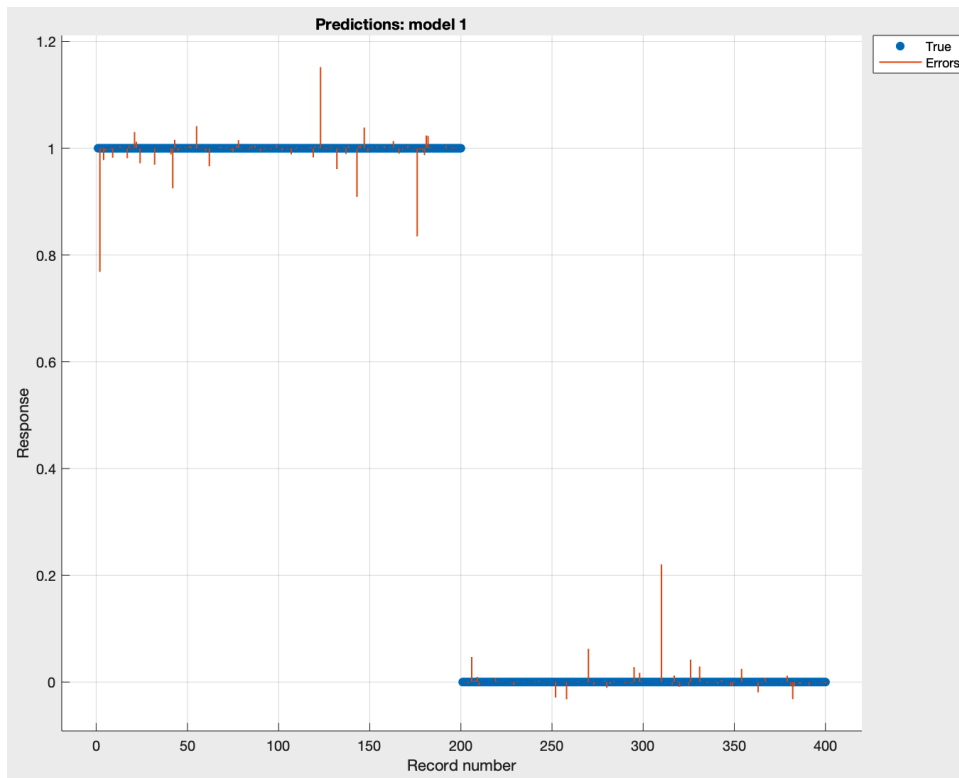


Figura 24: Relação entre valores preditos corretamente e incorretamente durante o treinamento do regressor logístico

5.3 Apresentação dos resultados de treinamento do *perceptron multicamadas*

A Figura 25 mostra o desempenho do erro quadrático médio durante o treinamento do *perceptron* multicamadas, nos três conjuntos: treinamento, validação e teste.

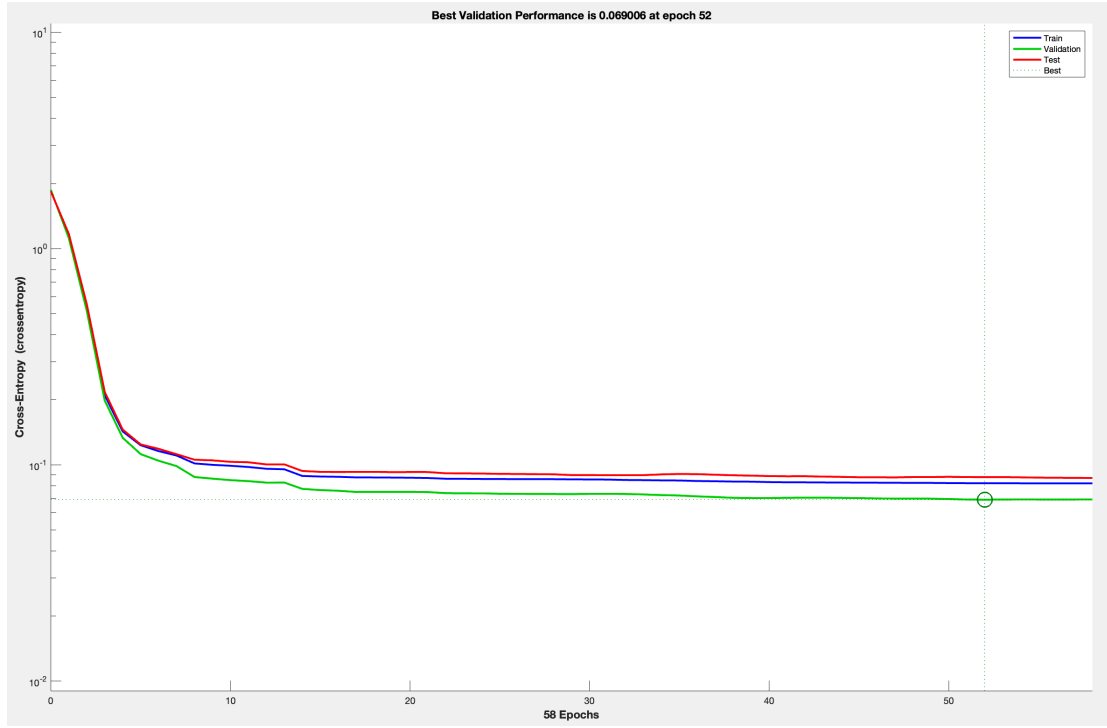


Figura 25: Desempenho do MSE durante o treinamento do *perceptron* multicamadas para detecção de parasitas

A melhor performance dessa rede é atingida na época 52 e o menor valor de MSE atingido foi de 0,0690.

5.4 Apresentação dos resultados de teste

A primeira tabela a ser apresentada, está referida na seção 4.2.1 Aplicação de Técnicas de PDI para Extração de ROIs. A Tabela 4 apresenta o desempenho da arquitetura *perceptron* simples com três tamanhos de filtros diferentes para remoção de WBCs.

Tabela 4 – Desempenho da arquitetura *Perceptron* simples com três tamanhos de filtros para remoção de WBCs

Tamanho do filtro de remoção de WBCs	Arquitetura	VP	VN	FP	FN	Precisão	Sensibilidade	F1-score
750	<i>Perceptron</i> simples	11378	-	910	1366	92,59%	89,28%	90,91%
1000		11485	-	910	1259	92,66%	90,12%	91,37%
1300		11485	-	2200	1259	83,92%	90,12%	86,91%

O desempenho das três arquiteturas de redes rasas utilizadas nesse trabalho é mostrado na Tabela 5. Conforme já citado, as métricas foram avaliadas considerando o conjunto de teste, que contém 15% das 1819 imagens testadas. Essas imagens contêm 12.744 parasitas.

Tabela 5 – Desempenho das arquiteturas

Arquitetura	Precisão	Sensibilidade	F1-score
<i>Perceptron</i> simples	92,66%	90,12%	91,37%
Regressor logístico	93,06%	94,87%	93,96%
<i>Perceptron</i> multicamadas	96,33%	98,85%	97,57%

As matrizes de confusão que geraram os resultados da Tabela 5 estão ilustradas nas Tabelas 6, 7 e 8, respectivamente.

Tabela 6 – Matriz de confusão para arquitetura *perceptron* simples

n = 12.744		Real	
		Positivo	Negativo
Predito	Parasitas	11485	1259
	Não-parasitas	910	-

Tabela 7 – Matriz de confusão para arquitetura do regressor logístico

n = 12.744		Real	
		Positivo	Negativo
Predito	Parasitas	12090	654
	Não-parasitas	902	-

Tabela 8 – Matriz de confusão para arquitetura *perceptron* multicamadas

n = 12.744		Real	
		Positivo	Negativo
Predito	Parasitas	12597	147
	Não-parasitas	480	-

A Tabela 9 apresenta uma comparação dos resultados obtidos com as arquiteturas propostas nesse trabalho com resultados obtidos em outros trabalhos anteriormente publicados na literatura e que utilizam a mesma base de dados.

Tabela 9 – Comparação dos resultados obtidos com a arquitetura 1 com outros resultados obtidos na literatura

Arquitetura	Classificação em	Tipo de imagem	Acurácia	Precisão	Sensibilidade	F1-score	Base de dados
<i>Perceptron</i> simples	<i>Pixels</i>	Gota Espessa	-	92,66%	90,12%	91,37%	<i>Chittagong Medical College Hospital, Bangladesh.</i>
Regressor logístico			-	93,06%	94,87%	93,96%	
<i>Perceptron</i> multicamadas			-	96,33%	98,85%	97,57%	
Yang et al. (2020)	<i>Patches</i>	Gota Espessa	97,26%	78,98%	82,73%	80,81%	
Aris et al. (2020)	<i>Pixels</i>	Gota Espessa	99,81%	-	55,24%	-	

Em relação a complexidade computacional, o método utilizando o *perceptron* simples leva 1,18s para análise de uma lâmina, enquanto que o regressor logístico e o *perceptron* multicamadas levam 1,2s e 3,29s, respectivamente.

5.5 Problemas e soluções

Esta seção visa apresentar problemas encontrados durante o desenvolvimento deste trabalho. O primeiro problema foi a detecção de objetos no centro ou ao redor da área de um WBC, que não eram parasitas, ou seja, eram FPs. A Figura 25(a) exemplifica uma detecção desse tipo. O círculo vermelho denota um falso positivo na região demarcada.

A solução adotada para que esse falso positivo não seja detectado foi a seguinte: após a detecção de uma célula WBC, traçamos um círculo em torno do mesmo, como mostrado na Figura 25(b), com um raio maior do que o raio da célula. A área circunscrita pelo círculo amarelo, denominada de região de interesse do WBC (*region of interest* ROI - WBC) é eliminada da inspeção. Dessa forma, para eliminação de um WBC e possíveis falsos positivos no seu entorno, adotamos o seguinte procedimento:

Pré-processamento → obtenção de WBCs → exclusão de ROI-WBCs → classificação

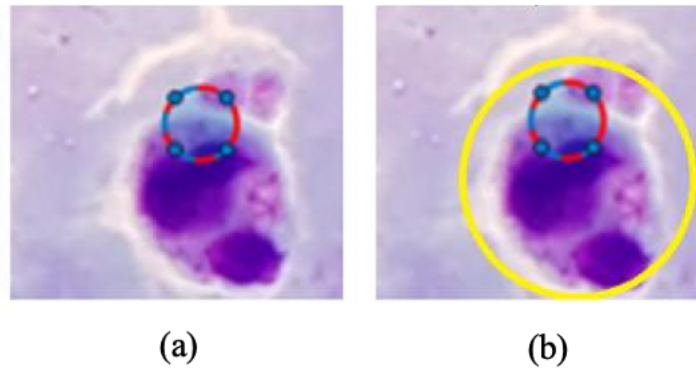


Figura 25: Exemplificação do problema. (a) FP; (b) região demarcada para que não seja inspecionada.

6 DISCUSSÃO

Este trabalho propôs um método para detecção e contagem de parasitas da malária em exames de sangue espesso, utilizando redes neurais rasas associadas a técnicas de processamento digital de imagens. O uso de técnicas de processamento de imagem para selecionar regiões de interesse permitiu que o classificador se concentrasse em um pequeno número de objetos, acelerando o processo de detecção do parasita da malária.

Diferentemente de outros trabalhos que utilizam redes neurais convolucionais complexas, como fast-RCNN (NUGROHO, 2021) e Yolov 3 (ABDURRAHMAN, 2021), este trabalho propõe classificadores simples para detecção do parasita da malária, como *perceptron*, regressão logística e *perceptron* multicamadas. Isso possibilita a automatização do método em dispositivos com baixa capacidade computacional, como *smartphones*.

A partir dos resultados apresentados nas Tabelas 5, 6, 7 e 8, observa-se que o *perceptron* multicamadas apresenta os melhores resultados. O *perceptron* multicamadas permitiu uma redução significativa de FP e FN em relação às outras redes. Registramos, no entanto que o regressor logístico e o *perceptron* simples são redes mais simples.

Destacamos que, na metodologia do trabalho, para a maioria dos processos que utilizam limiares, procurou-se utilizar métodos ótimos ou justificar o valor do limiar utilizado, através de uma exploração de valores. Por exemplo, no processo de binarização da imagem, utilizou-se o método ótimo de Otsu. Já no processo de filtragem das células WBCs, exploraram-se três valores distintos para o tamanho do filtro de área. Apenas no processo de eliminação de ruídos utilizou-se um limiar determinado experimentalmente.

Analisando os resultados apresentados na Tabela 9, percebe-se que o método de Aris et al. (2020) possui uma acurácia maior em relação aos outros trabalhos. No entanto, esse trabalho apresenta uma sensibilidade extremamente baixa, de 55,24%. A princípio, não entendemos como o autor reporta uma acurácia de 99,81%, com um valor de sensibilidade tão baixo.

Em comparação ao trabalho de Yang et al (2020), os métodos propostos nesse trabalho conseguiram um melhor equilíbrio entre sensibilidade (*recall*) e precisão, expresso pela métrica *F1-score*.

Em relação à complexidade computacional, observamos também que enquanto no trabalho de Yang et al.¹⁴ os autores relatam um tempo de 10s para análise de uma lâmina, o método proposto neste trabalho que utiliza o *perceptron* simples alcançou um tempo de 1,18s. Métodos que utilizam redes de aprendizagem profunda para detecção de objetos, como *fast-RCNN* (NUGROHO, 2021) e *Yolov 3* (ABDURRAHMAN, 2021), relatam pequenas complexidades de tempo, 250ms e 34ms, respectivamente. No entanto, esses tempos foram obtidos ao analisar imagens pequenas, de 1024 x 768 *pixels*. Neste trabalho, o tamanho da imagem analisada foi de 3024 × 4032 *pixels*.

7 CONCLUSÕES E TRABALHO FUTUROS

Esse Trabalho apresentou um método que, embora utilizando ferramentas simples de reconhecimento de padrões, apresentou resultados que o colocam no estado da arte em termos de desempenho e de complexidade computacional. Considerando a alta incidência da malária na região, a oportunidade de desenvolvimento de uma ferramenta computacional dessa natureza é de extrema utilidade, principalmente se a mesma for disponibilizada nas nuvens e puder ser acessada por cidades distantes dos grandes centros, ou ser disponibilizada para sistemas operacionais de dispositivos móveis.

8 REFERÊNCIAS

A. W. Setiawan, A. Faisal, N. Resfita and Y. A. Raman, "Detection of Malaria Parasites using Thresholding in rGB, YCbCr and Lab Color Spaces," 2021 International Seminar on Application for Technology of Information and Communication (iSemantic), 2021, pp. 70-75, doi: 10.1109/iSemantic52711.2021.9573224.

Barbosa, Luciana & Dantas, Silvrano & Anjos, Gerson. (2014). Modelo de Previsão de Recalques em Estacas Hélice Contínua Utilizando Redes Neurais Artificiais.

Brasil. (2020). Bases do Diagnóstico Microscópico da Malária. Parte 1 e Parte 2: Guia do Aluno. Brasília, D.F.: Organização Pan-Americana da Saúde. Licença: CC BY-NC-SA 3.0 IGO.

Bishop, C. M. Pattern Recognition and Machine Learning (Information Science and Statistics). Berlin, Heidelberg: Springer-Verlag, 2006.

C. B. Delahunt et al., "Automated microscopy and machine learning for expert-level malaria field diagnosis," in Proc. IEEE Global Humanitarian Technol. Conf., Seattle, WA, USA, 2015, pp. 393–399.

C. Mehanian, M. Jaiswal, C. Delahunt, and C. Thompson, "Computer-automated malaria diagnosis and quantitation using convolutional neural networks," in Proc. IEEE Int. Conf. Comput. Vision Workshops, Venice, Italy, 2017, pp. 116–125.

Communications, Signal Processing and Networking, WiSPNET 2017, 2017, vol. 2018-Janua, pp. 1303–1307, doi:10.1109/WiSPNET.2017.8299974.

Cybenco. "Continuous valued neural networks with two hidden layers are sufficient. Technical report. Department of Computer Sciences, Tufts University, Medford, Massachusetts, USA, 1988.

D. Fletcher, E. Goss, Forecasting with neural networks: an application using bankruptcy data, *Information and Management* 24 (3) (1993) 159 – 167.

Edward R. Dougherty. 1992. *An Introduction to Morphological Image Processing*. SPIE Optical Engineering Press., USA.

F. Abdurahman, K. A. Fante, and M. Aliy, “Malaria parasite detection in thick blood smear microscopic images using modified YOLOV3 and YOLOV4 models,” *BMC Bioinformatics*, 22(1), 2021, doi:10.1186/s12859-021-04036-4.

F. Yang et al., “Deep Learning for Smartphone-Based Malaria Parasite Detection in Thick Blood Smears,” *IEEE J. Biomed. Heal. Informatics*, 24(5), pp. 1427–1438, 2020, doi:10.1109/JBHI.2019.2939121.

GÉRON, A. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. [s.l.] O’Reilly Media, 2017b.

H. A. Nugroho and R. Nurfuzi, "Deep Learning Approach for Malaria Parasite Detection in Thick Blood Smear Images," 2021 17th International Conference on Quality in Research (QIR): International Symposium on Electrical and Computer Engineering, 2021, pp. 114-118, doi: 10.1109/QIR54354.2021.9716198.

Hagan, M. T., Demuth, H. B., Beale, M. H. and Jesús, O. De., “Neural Network Design, 2nd ed.”, Martin T. Hagan and Howard B. Demuth, Frisco (2020).

I. R. Dave and K. P. Upla, “Computer aided diagnosis of Malaria disease for thin and thick blood smear microscopic images,” in: 2017 4th International Conference on Signal Processing and Integrated Networks, SPIN 2017, 2017, pp. 561–565, doi: 10.1109/SPIN.2017.8050013.

I. R. Dave, “Image analysis for malaria parasite detection from microscopic images of thick blood smear,” in: *Proceedings of the 2017 International Conference on Wireless*

J. Quinn, A. Andama, I. Munabi, F. Kiwanuka. Automated blood smear analysis form of malaria diagnosis; 2014. P. 115-132. ISBN 978-1-4665-8929-2.

Manual de diagnóstico laboratorial da malária / Ministério da Saúde, Secretaria de Vigilância em Saúde – 2. ed. – Brasília : Ministério da Saúde, 2009.

Mathworks. Elementos estruturantes. Disponível em: <<https://www.mathworks.com/help/images/structuring-elements.html>>. Acesso em: 6 outubro 2022.

Mundo Educação. Disponível em: <mundoeducacao.uol.com.br/biologia/neuronios.html>. Acesso em: 21 out. 2022.

Nugroho, H. A. and Nurfauzi, R., “Deep Learning Approach for Malaria Parasite Detection in Thick Blood Smear Images,” 17th Int. Conf. Qual. Res. QIR 2021 Int. Symp. Electr. Comput. Eng., 114–118 (2021).

PATTERSON, J.; GIBSON, A. Deep Learning: A Practitioner’s Approach, 2017.

Rajaraman S, Antani SK, Poostchi M, Silamut K, Hossain MA, Maude RJ, Jaeger S, Thoma GR. 2018. Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images. PeerJ 6:e4568 <https://doi.org/10.7717/peerj.4568>.

Rafael C. Gonzalez and Richard E. Woods. 2006. Digital Image Processing (3rd Edition). Prentice-Hall, Inc., USA.

S. R. Abidin, U. Salamah, and A. S. Nugroho, “Segmentation of malaria parasite candidates from thick blood smear microphotographs image using active contour without edge,” 2016, doi:10.1109/IBIOMED.2016.7869824.

T. A. Aris, A. S. A. Nasir, L. C. Chin, H. Jaafar, and Z. Mohamed, “Fast k-means clustering algorithm for malaria detection in thick blood smear,” in: 2020 IEEE 10th International

Conference on System Engineering and Technology, ICSET 2020 - Proceedings, 2020, pp. 267–272, doi:10.1109/ICSET51301.2020.9265380.

World Health Organization, World health statistics 2021: monitoring health for the SDGs, sustainable development goals. 2021.

Walczak, Steven and Cerpa, Narciso, "Heuristic Principles for the Design of Artificial Neural Networks" (1999). School of Information Faculty Publications. 211.

Young, Ian. (1983). Image analysis and mathematical morphology, by J. Serra. Academic Press, London, 1982. Cytometry. 4. 184-185. 10.1002/cyto.990040213.