

**UNIVERSIDADE FEDERAL DE ALAGOAS-UFAL
INSTITUTO DE COMPUTAÇÃO
CIÊNCIA DA COMPUTAÇÃO**

EDUARDO ANTÔNIO DE LUCENA LISBOA

**AVALIAÇÃO DE MÉTODOS CLÁSSICOS DE DETECÇÃO DE CARACTERÍSTICAS
NO AUXÍLIO À IDENTIFICAÇÃO DE AMASTIGOTAS DE LEISHMANIOSE**

**MACEIÓ - AL
2023**

Eduardo Antônio de Lucena Lisboa

Avaliação de Métodos Clássicos de Detecção de Características no Auxílio à Identificação de
Amastigotas de Leishmaniose

Monografia apresentada como requisito parcial
para obtenção do grau de Bacharel em Ciência da
Computação da Universidade Federal de Alagoas -
UFAL, Instituto de Computação.

Orientador: Prof^ª. Dra. Fabiane da Silva
Queiroz

Maceió - AL

2023

Catálogo na Fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico

Bibliotecário: Marcelino de Carvalho Freitas Neto – CRB-4 - 1767

L769a Lisboa, Eduardo Antônio de Lucena.

Avaliação de métodos clássicos de detecção de características no auxílio à identificação de amastigotas de leishmaniose / Eduardo Antônio de Lucena Lisboa. – 2023.
47 f. : il.

Orientador: Fabiane da Silva Queiroz.

Monografia (Trabalho de conclusão de curso em Ciência da Computação) – Universidade Federal de Alagoas, Instituto de Computação. Maceió, 2023.

Bibliografia: f. 43-47.

1. Leishmaniose visceral. 2. Aprendizagem de máquina. 3. Diagnóstico médico. I. Título.

CDU: 004.81:159.953.5:616.928.5

Dedico este trabalho aos meus pais e à minha
noiva.

AGRADECIMENTOS

Agradeço aos meus pais pela educação e oportunidades que eles me deram e dão. Me tornando um homem melhor a cada dia.

Agradeço profundamente a minha noiva, Maria. Sem ela os dias com a cara enfiada no computador teriam sido bem mais difíceis.

Agradeço aos meus amigos, que me absterei de nomear pois certamente me esqueceria de algum, por todos os encontros para comer, jogar ou apenas conversar e rir até horas esquecidas.

Agradeço minha orientadora, Fabiane, pela paciência infinda para comigo.

Agradeço ao Programa de Pesquisa para o SUS (PPSUS) pelo financiamento da bolsa de pesquisa, contribuindo para a realização deste trabalho.

E por fim agradeço a Sony e a Nintendo por fazerem consoles tão eficientes em nos distrair.

E aqueles que foram vistos dançando foram julgados insanos por aqueles que não podiam escutar a música.

Friedrich Nietzsche

RESUMO

A Leishmaniose Visceral (LV) é uma antroponose, de alta letalidade, transmitida pelo popularmente conhecido *mosquito palha*. Atualmente, o diagnóstico dessa doença é complexo devido à similaridade de seus sintomas com outras enfermidades, sendo o padrão ouro do diagnóstico o exame parasitológico que trata da análise visual do parasita *leishmania* em material de biópsia, sendo esta análise passível de erros devido a fatores como o tamanho minúsculo do parasita a ser observado (cerca 2–4 μm de diâmetro) e a expertise médica no diagnóstico dessa doença. Neste estudo, a partir de um método *patch-based*, avaliamos diversos descritores de textura e algoritmos de classificação que possam auxiliar na identificação destes parasitas. Mais precisamente, um conjunto de fragmentos (*patches*), extraídos de imagens microscópicas de aspirado de medula óssea, foram submetidos a algoritmos de pré-processamento, extração de características e classificação. Os algoritmos SIFT, ORB e Haralick foram empregados na extração de características, considerando sua detecção, descrição e análise de textura. Os classificadores SVM e KNN foram utilizados para classificar as imagens, identificando a presença ou ausência de *leishmania*. Os resultados obtidos demonstraram a eficácia dessas técnicas no diagnóstico da LV, com acurácia satisfatória na identificação dos parasitas. Portanto, este estudo destaca o potencial das técnicas de processamento de imagens e algoritmos de aprendizagem de máquina como ferramentas auxiliares no diagnóstico preciso e rápido dessa doença grave, possibilitando um tratamento eficaz e melhores resultados para os pacientes afetados.

Palavras-chave: Leishmaniose visceral humana. Extração de características. Aprendizagem de máquina. Classificação. Diagnóstico médico.

ABSTRACT

Visceral leishmaniasis (VL) is an anthroponosis, of high lethality, transmitted by the popularly known *sand flies*. Currently, the diagnosis of this disease is complex due to the similarity of its symptoms with other diseases, the gold standard of diagnosis being the parasitological examination that deals with the visual analysis of the parasite *leishmania* in biopsy material, this analysis being subject to errors due to factors such as the minuscule size of the parasite to be observed (about 2–4 μm in diameter) and the medical expertise in diagnosing this disease. In this study, from a *patch-based* method, we evaluated several texture descriptors and classification algorithms that can help identify these parasites. More precisely, a set of fragments (*patches*), extracted from microscopic images of bone marrow aspirates, were submitted to pre-processing, feature extraction and classification algorithms. The algorithms SIFT, ORB and Haralick were employed in the feature extraction, considering their detection, description and texture analysis. The classifiers SVM and KNN were used to classify the images, identifying the presence or absence of *leishmania*. The results obtained demonstrated the effectiveness of these techniques in the diagnosis of VL, with satisfactory accuracy in the identification of parasites. Therefore, this study highlights the potential of image processing techniques and machine learning algorithms as auxiliary tools in the accurate and rapid diagnosis of this serious disease, enabling effective treatment and better outcomes for affected patients.

Keywords: Human visceral leishmaniasis. Feature extraction. Machine learning. Classification. Medical diagnostics.

LISTA DE FIGURAS

Figura 1 – Ciclo de Contaminação da LV	14
Figura 2 – Exemplo <i>Dataset 1</i>	18
Figura 3 – Exemplo <i>Dataset 2</i>	19
Figura 4 – Pré-processamento dos Rótulos - <i>Dataset 1</i>	20
Figura 5 – Passos de Pré-processamento das Imagens - <i>Dataset 2</i>	21
Figura 6 – Curvas ROC - <i>Dataset 1</i>	30
Figura 7 – Curvas ROC - <i>Dataset 1</i> Pré-processado	32
Figura 8 – Curvas ROC - <i>Dataset 2</i>	34
Figura 9 – Curvas ROC - <i>Dataset 2</i> Aumentado	36

LISTA DE TABELAS

Tabela 1 – Dataset 1 - SVM	29
Tabela 2 – Dataset 1 - KNN	30
Tabela 3 – Dataset 1 Pré-processado - SVM	31
Tabela 4 – Dataset 1 Pré-processado - KNN	31
Tabela 5 – Dataset 2 - SVM	33
Tabela 6 – Dataset 2 - KNN	33
Tabela 7 – Dataset 2 Aumentado - SVM	35
Tabela 8 – Dataset 2 Aumentado - KNN	35
Tabela 9 – SIFT - Dataset 1	37
Tabela 10 – SIFT - Dataset 1 Pré-processado	37
Tabela 11 – SIFT - Dataset 2	37
Tabela 12 – SIFT - Dataset 2 Aumentado	37
Tabela 13 – ORB - Dataset 1	38
Tabela 14 – ORB - Dataset 1 Pré-processado	38
Tabela 15 – ORB - Dataset 2	38
Tabela 16 – ORB - Dataset 2 Aumentado	38
Tabela 17 – Haralick - Dataset 1	39
Tabela 18 – Haralick - Dataset 1 Pré-processado	39
Tabela 19 – Haralick - Dataset 2	39
Tabela 20 – Haralick - Dataset 2 Aumentado	39

LISTA DE ABREVIATURAS E SIGLAS

AM	Aprendizagem de Máquina
AUC	Area Under Curve
AVC	Acidente Vascular Cerebral
BRIEF	Binary Robust Independent Elementary Features
FAST	Features from Accelerated Segment Test
GCLM	Gray-Level Co-occurrence Matrix
KNN	K-Nearest Neighbors
LV	Leishmaniose Visceral
OMS	Organização Mundial de Saúde
ORB	Oriented FAST and Rotated BRIEF
ROC	Receiver Operating Characteristic
SIFT	Scale-invariant feature transform
SVM	Support Vector Machine

LISTA DE SÍMBOLOS

μ	Letra grega Mu
σ	Letra grega Sigma

SUMÁRIO

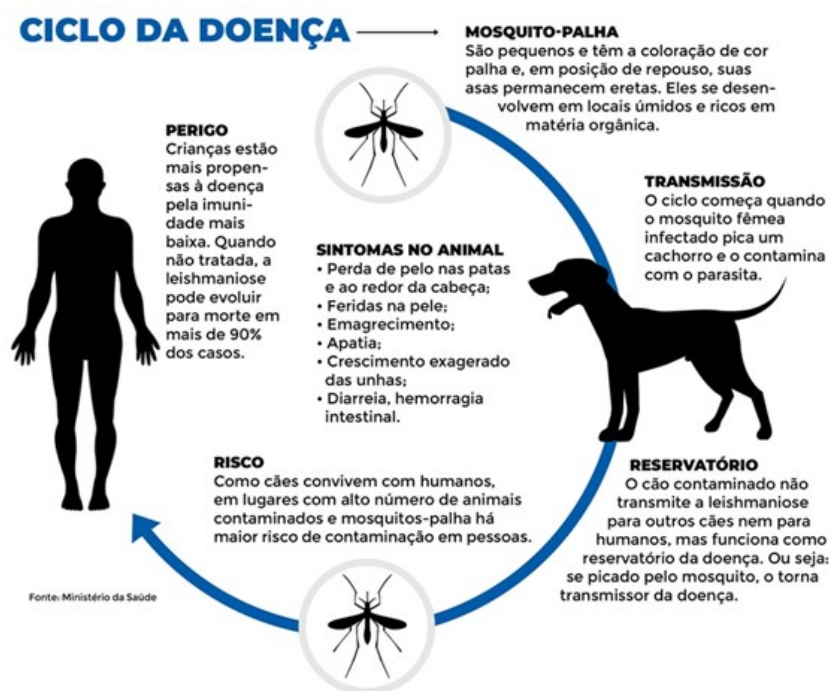
1	INTRODUÇÃO	14
2	FUNDAMENTAÇÃO TEÓRICA	16
2.1	PROCESSAMENTO DE IMAGENS	16
2.2	APRENDIZAGEM DE MÁQUINA NO AUXÍLIO AO DIAGNÓSTICO MÉDICO	16
3	MÉTODO	18
3.1	PRÉ-PROCESSAMENTO	19
3.1.1	Pré-processamento de imagens microscópicas	19
3.1.2	Extração da área de interesse	21
3.1.3	Extração de <i>patches</i> das imagens	21
3.1.4	Amostragem de dados	22
3.2	Extração de características	22
3.3	Classificação	27
3.4	Métricas de avaliação	27
4	RESULTADOS E DISCUSSÕES	29
4.1	RESULTADOS	29
4.1.1	<i>Dataset 1</i>	29
4.1.2	<i>Dataset 1</i> Pré-processado	31
4.1.3	<i>Dataset 2</i>	32
4.1.4	<i>Dataset 2</i> Aumentado	34
4.1.5	SIFT	36
4.1.6	ORB	37
4.1.7	Haralick	38
4.2	DISCUSSÃO	39
5	CONCLUSÃO	42
	REFERÊNCIAS	43

1 INTRODUÇÃO

A Leishmaniose Visceral (LV) humana é uma doença grave causada pelo protozoário *Leishmania chagasi* e transmitida pela picada do mosquito palha, também conhecido como asa-dura, tatuquiras, birigui, dentre outros. Se não tratada, pode levar a óbito até 90% dos casos (BRASIL, 2022a). Por ser considerada uma "doença tropical negligenciada" que afeta diversos países, a Organização Mundial de Saúde (OMS), atualmente está com um projeto para erradicá-las até 2030 (OMS, 2020).

A transmissão acontece quando fêmeas do mosquito infectadas picam cães ou outros animais infectados, e depois picam o homem, transmitindo o protozoário *Leishmania chagasi*, causador da LV (BRASIL, 2019). O ciclo pode ser visualizado na Figura 1.

Figura 1 – Ciclo de Contaminação da LV



Fonte: Ministério da Saúde (2019)

A LV é uma doença endêmica em 76 países (na Ásia, na Europa, no Oriente Médio, na África e nas Américas), sendo o Brasil responsável por 90% dos casos na América Latina e, até 2009, seu número de mortes só era superado pela malária (BRASIL, 2022b; ALVES, 2009). O diagnóstico da doença é complexo e trabalhoso devido à semelhança de seus sintomas com outras doenças como febre tifoide, tuberculose e malária, podendo ocorrer coinfeções dessas doenças junto com a LV (KUMAR et al., 2020).

A LV é também considerada emergente em indivíduos portadores da infecção pelo vírus da imunodeficiência adquirida (HIV), tornando-se uma das doenças mais importantes da

atualidade (BRASIL, 2006).

O padrão ouro para o diagnóstico da LV é o exame parasitológico, que consta da identificação visual das amastigotas¹ de *leishmania* em material de biópsia de fígado, baço, nódulos linfáticos ou medula óssea. No entanto, essa forma de diagnóstico é propensa à erros, pois as amastigotas, além de possuírem um tamanho minúsculo (cerca de 2-4 μm), se assemelham a diversas outras estruturas celulares, portanto, é necessário contar com profissionais experientes para um diagnóstico correto (CARNEIRO et al., 2011).

O procedimento mais seguro é obter a biópsia da medula óssea, e a sensibilidade da medula óssea corada com *Giemsa*² é de cerca de 60% a 85% (ELMAHALLAWY et al., 2014).

Com os avanços na computação e nas tecnologias e técnicas de Processamento de Imagens e métodos para extração de características em imagens, como os algoritmos SIFT, ORB e Haralick, o potencial dessas tecnologias serem implementadas na análise de imagens em laboratórios e hospitais, junto com profissionais avaliadores, tem crescido exponencialmente (LOWE, 1999; RUBLEE et al., 2011; HARALICK et al., 1973).

Essas tecnologias, aliadas a algoritmos de aprendizagem de máquina, mostram-se extremamente eficazes para o auxílio ao diagnóstico médico, inclusive já sendo testadas e implementadas em diversas áreas médicas, como identificação de câncer de mama (ROY et al., 2021), reconstrução da textura da cavidade abdominal (WU et al., 2022), detecção de glaucoma (SINGH et al., 2022) entre diversas outras áreas (CUI et al., 2017; TAHIR et al., 2005; MONDAL et al., 2015).

Considerando os avanços nos algoritmos de extração de características e de aprendizagem de máquina, bem como a dificuldade na identificação visual de amastigotas de leishmaniose, este trabalho propõe avaliar alguns dos algoritmos mais comuns mencionados anteriormente, juntamente com os classificadores SVM e KNN, na identificação de características e classificação de imagens de microscópio para a presença de amastigotas de LV.

Este documento está estruturado da seguinte forma: O Capítulo 2 apresenta uma fundamentação teórica dos conceitos aqui utilizados; O Capítulo 3 apresenta os métodos utilizados para a execução deste trabalho; O Capítulo 4 apresenta os resultados obtidos da utilização do método exposto no Capítulo 3 e discute tais resultados comparando-os com a literatura; E, por fim, o Capítulo 5 apresenta nossas conclusões e discussão sobre trabalhos futuros.

¹ Amastigotas são células arredondadas de protozoários sem flagelo.

² A solução de coloração de *Giemsa* é uma das manchas microscópicas mais comuns, geralmente usada em hematologia, histologia, citologia e bacteriologia para diagnóstico in vitro.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 PROCESSAMENTO DE IMAGENS

Processamento de imagens é um campo da ciência da computação que lida com a manipulação de imagens digitais. Técnicas de processamento podem ser utilizadas para aumentar a qualidade das mesmas, extrair características e para classificá-las, entre outras aplicações (GONZALEZ; WOODS, 2002; KAUFFMAN, 2007; DOUGHERTY, 2014).

Existem diversas técnicas de processamento de imagens que já foram utilizadas para o diagnóstico da LV. Uma técnica comum é usar um microscópio para aumentar a imagem de aspirado de nódulo linfático ou raspagem de pele. Isto permite que o examinador identifique características morfológicas dos parasitas de *leishmania* (FUENTE; ARENAS, 2008).

Outra técnica que tem sido utilizada para o diagnóstico é a de manchar a imagem de aspirado de nódulo linfático com um corante como o *Giemsa*, que é um corante que contém uma mistura de outros corantes que se ligam a diferentes componentes das células, resultando em um padrão de coloração característico. Esta técnica de coloração torna os parasitas da leishmania mais visíveis e fáceis de identificar (HERRERO; MORENO, 2010).

A utilização do processamento de imagens em imagens médicas e diagnósticos está crescendo constantemente. Uma aplicação é a detecção de células cancerígenas em imagens médicas como mamografias e exames de ressonância magnética. Além disso, as técnicas de processamento de imagens podem auxiliar em procedimentos cirúrgicos e auxiliar no planejamento de tratamento (CHEN et al., 2020; RAJKOMAR et al., 2019b).

O processamento de imagens também está sendo usado para diagnosticar a LV, uma doença grave que pode ser fatal se não for tratada. O processamento de imagens pode ser usado para detectar a LV identificando características da doença em imagens médicas, como baços e fígados aumentados (SINGH et al., 2018).

2.2 APRENDIZAGEM DE MÁQUINA NO AUXÍLIO AO DIAGNÓSTICO MÉDICO

Aprendizagem de Máquina é um tipo de inteligência artificial que permite computadores a aprender sem serem explicitamente programados. Algoritmos de AM são treinados com dados e eles podem ser utilizados para realizar previsões ou decisões sobre novos dados (MURPHY, 2012; HASTIE et al., 2009; BURKOV, 2019).

AM vem sendo utilizada em uma variedade de aplicações médicas, incluindo o diagnóstico médico. Algoritmos de AM podem ser utilizados para analisar imagens médicas como raios-x, ressonâncias magnéticas e tomografias computadorizadas. Eles também podem ser utilizados para analisar dados médicos, como resultados de exames de sangue e o histórico do paciente (KHORASANI et al., 2019; ZHANG et al., 2020).

Tais algoritmos podem ser usados para melhorar a acurácia de diagnósticos médicos. Por exemplo, algoritmos de AM já demonstraram ser mais precisos do que especialistas humanos no diagnóstico de câncer de pele a partir de imagens de lesões de pele (ESTEVA et al., 2017; LEE et al., 2019; SHIN et al., 2018).

AM é uma tecnologia promissora que tem potencial de revolucionar os diagnósticos médicos. Estes algoritmos podem auxiliar médicos a terem diagnósticos mais precisos e a identificar pacientes que estão em risco de desenvolver certas doenças. Isto pode levar a diagnósticos precoces e tratamento de doenças, melhorando os resultados dos pacientes (GHOSH et al., 2020; RAJKOMAR et al., 2019a).

Nos últimos anos, houve um desenvolvimento significativo de algoritmos de AM aplicados ao campo médico, demonstrando capacidade de superar especialistas humanos no diagnóstico de várias doenças (BUSH et al., 2020; BLEI et al., 2020; LEIBOWITZ, 2020). Em 2017, o Google AI desenvolveu um algoritmo com 90% de acurácia na detecção de retinopatia diabética em exames oculares, equiparando-se a oftalmologistas.

Na dermatologia, pesquisadores da Universidade da Pensilvânia criaram um algoritmo, em 2018, que alcançou uma acurácia de 95% na detecção de câncer de pele por meio de imagens de lesões cutâneas (GULATI et al., 2017; ESTEVA et al., 2017). Em 2019, pesquisadores da Universidade da Califórnia desenvolveram um algoritmo com 80% de acurácia na predição de ataques cardíacos e AVCs, superando os modelos usados pelos médicos atualmente (RAJKOMAR et al., 2019b).

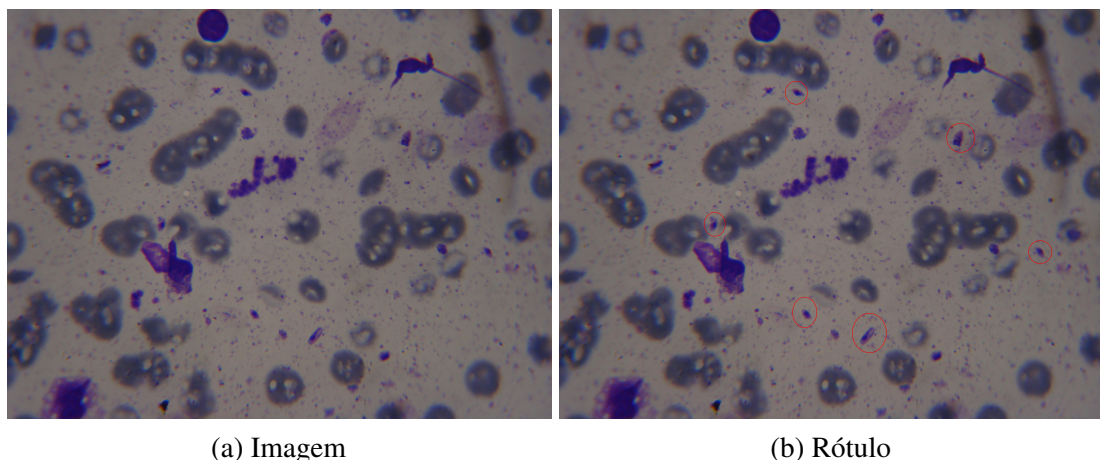
3 MÉTODO

Foram utilizados duas bases de imagens, doravante chamados *Datasets*. O primeiro é um *Dataset* indiano¹ contendo um total de 45 pares de imagens coloridas de microscópio de aspirado de medula óssea, captados utilizando uma câmera digital (Sony DSC_H9) acoplada a um microscópio óptico (Olympus-CH40RF200) (FARAHI et al., 2014). Os pares (Figura 2) são imagens iguais sendo a primeira uma imagem sem marcações (Figura 2a) e a outra é a mesma imagem mas com uma marcação (círculo vermelho) do local onde há uma ou mais amastigotas de *leishmania* identificados (Figura 2b).

O segundo *Dataset* consiste em 68 pares de imagens coloridas de microscópio, captados utilizando um celular (iPhone 8) e um microscópio óptico com aumento de 1000x (MARINHO, 2020). Cada um dos pares (3), é composto pela imagem capturada (Figura 3a) e uma imagem binária (preta e branca) de mesmas dimensões da imagem original com o fundo preto e as regiões onde foram identificados amastigotas de LV marcadas com um quadrado branco de tamanho 50x50 *pixels* (Figura 3b).

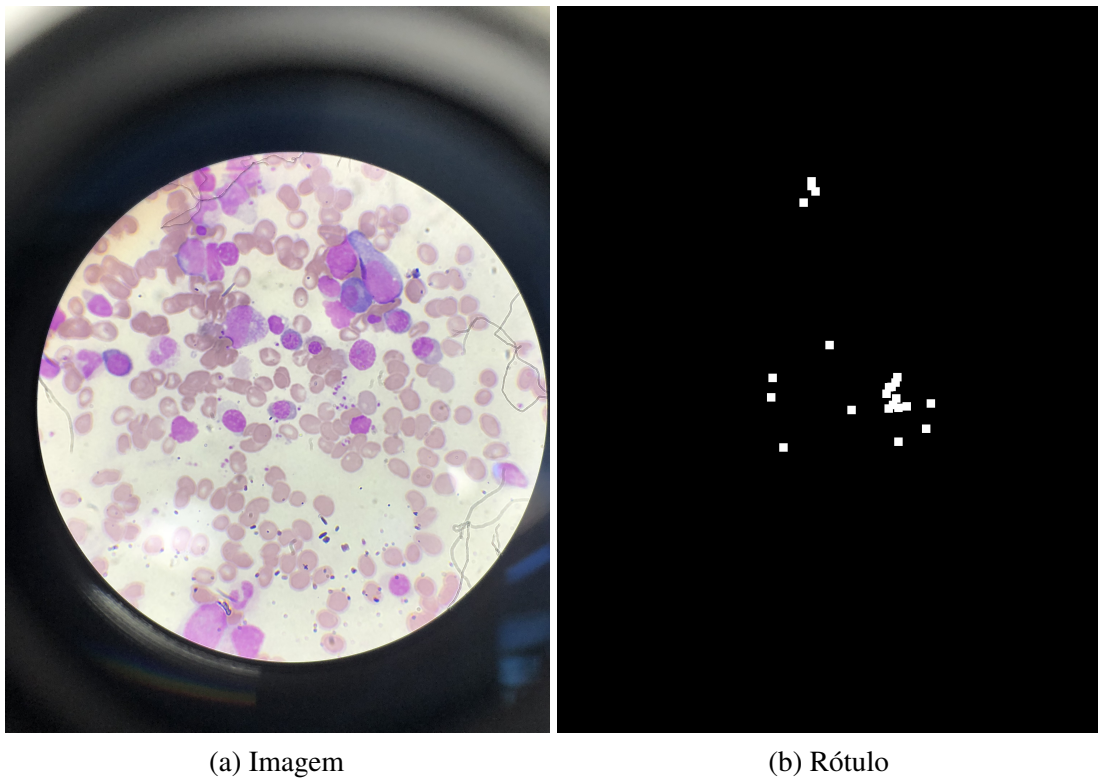
Para desenvolver nosso método de avaliação, utilizamos a linguagem de programação Python na sua versão 3.10.7 e as bibliotecas *Scikit-Learn* para os classificadores e métricas de avaliação, a biblioteca *Matplotlib* para os gráficos, a biblioteca *Mahotas* para o extrator de características Haralick e a biblioteca *OpenCV* para os demais extratores de características e demais manipulações das imagens (PEDREGOSA et al., 2011; HUNTER, 2007; COELHO, 2013; ITSEEZ, 2015).

Figura 2 – Exemplo *Dataset* 1



Fonte: Fahari (2014)

¹ Disponível em: <<https://sites.google.com/site/hosseinrabbanikhorasgani/datasets-1/dataset-of-leishmania-parasite-in-microscopic-images>>

Figura 3 – Exemplo *Dataset 2*

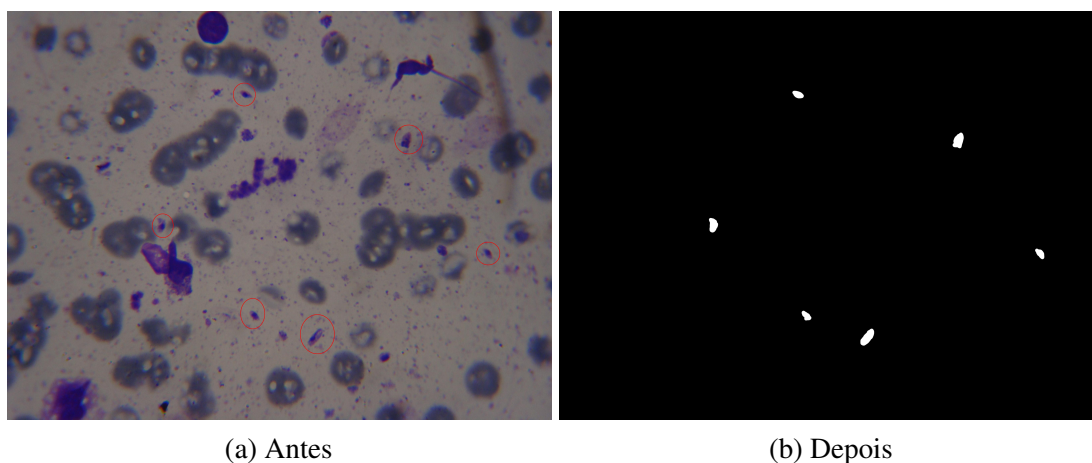
Fonte: Marinho (2020)

3.1 PRÉ-PROCESSAMENTO

3.1.1 Pré-processamento de imagens microscópicas

No *Dataset 1* não foi necessário um pré-processamento das imagens sem marcação pois as mesmas não possuem a área exterior do microscópio ou outros elementos presentes que possam interferir nos modelos e algoritmos utilizados como é possível ver na Figura 2a, mas as imagens com marcação foram processadas de maneira a tornar o rótulo uma imagem binária, tal qual a Figura 3b do *Dataset 2*.

A rotulação manual dessas imagens foi feita utilizando o programa de computador *Photoshop* (Adobe Inc., 2023). Cada imagem com marcação foi inserida, uma nova imagem totalmente preta foi inserida por cima e, reduzindo sua opacidade, foi pintada de branco onde havia uma marcação de amastigota na imagem com marcação e então essa nova imagem criada foi salva como uma nova imagem. O resultado está presente na Figura 4.

Figura 4 – Pré-processamento dos Rótulos - *Dataset 1*

Fonte: Autor desta monografia, 2023.

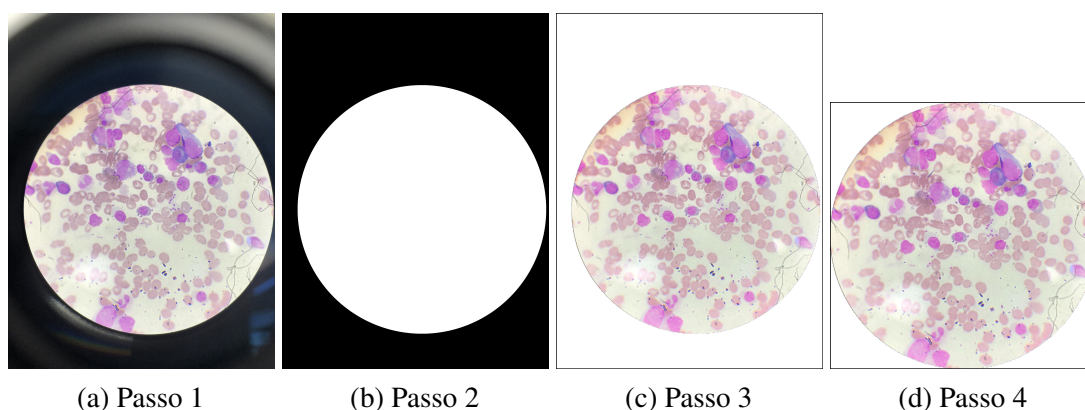
Já no *Dataset 2*, foi necessário um pré-processamento um pouco maior pois a área exterior do microscópio é visível na imagem, como é possível ver na Figura 3a. Como nosso método é uma análise baseada em fragmentos (*patch-based*), manter essa área geraria muitos *patches* que atrapalhariam os algoritmos, prejudicando a classificação do conjunto total de *patches* da imagem.

O pré-processamento foi realizado através dos seguintes passos:

1. Recortar a área do exterior do microscópio;
2. Encontrar o círculo que melhor engloba todo o microscópio utilizando o algoritmo *Hough Circles*;
3. Criar uma imagem binária com o círculo em branco e o fundo em preto (Figura 5b);
4. Encontrar o menor quadrado que circunscreve este círculo, tornando branco tudo que não está dentro do círculo (Figura 5c); e
5. Cortar a imagem original neste quadrado (Figura 5d).

O processo pode ser visualizado na Figura 5. As bordas nas Figuras 5c e 5d foram adicionadas apenas para facilitação da visualização.

Figura 5 – Passos de Pré-processamento das Imagens - *Dataset 2*



Fonte: Autor desta monografia, 2023.

3.1.2 Extração da área de interesse

Após os pré-processamentos, utilizando as imagens rótulo como referência, de cada imagem de cada *Dataset* foram extraídos *patches* de tamanho 70x70 contendo as amastigotas de *leishmania*. Estes *patches* serão utilizados como rótulos no processo de classificação.

3.1.3 Extração de *patches* das imagens

Ambos os *Datasets* foram submetidos à um processo de retalhamento utilizando uma grade regular. A partir de cada imagem foi gerado um conjunto de *patches* de tamanho 70x70.

Pra gerar *patches* a serem usados como rótulos da classe de interesse (verdadeiros positivos), ou seja, *patches* que contenham o parasita, nós utilizamos os rótulos descritos na Seção 3.1.1 para retirar da imagem original o *patch* contendo a amastigota e, em seguida, substituindo-o por um *patch* branco.

Considerando que estamos tratando de uma classificação binária, para gerar *patches* da segunda classe (verdadeiros negativos), nós utilizamos as imagens resultantes da extração dos verdadeiros positivos, onde uma grade regular foi posta sobre a imagem e cada célula da grade foi extraída como um *patch* da segunda classe, descartando quaisquer *patches* que tivessem a presença de um ou mais *pixels* branco.

O *Dataset 1*, que possuía 45 imagens, tornou-se um *Dataset* com 91 *patches* contendo amastigotas e 4526 *patches* sem amastigotas. Já o *Dataset 2* passou de 68 imagens para 760 *patches* com e 76.495 *patches* sem amastigotas. A partir disso, podemos perceber que os novos *datasets* estão extremamente desbalanceados - aproximadamente 1:50 no *Dataset 1* e 1:100 no *Dataset 2*.

Isso ocorre justamente pelo fato de os objetos da classe de interesse (amastigotas de *leishmania*) apresentarem-se em pequeno número e de forma minúscula nas imagens analisadas. Devido a este desbalanceamento, um processo de amostragem sobre os dados precisou ser executado antes das etapas de extração de características e classificação. Na Seção 3.1.4 discutimos mais detalhadamente como essa amostragem foi conduzida.

3.1.4 Amostragem de dados

Como os *Datasets* apresentaram um desbalanceamento muito grande, com a proporção de *patches* com amastigotas (verdadeiros positivos) para *patches* sem amastigotas (verdadeiros negativos), nós utilizamos duas formas de amostragem sobre os dados:

- Um *Oversampling* sobre classe de interesse utilizando recursos de *data augmentation*
- Um *Undersampling* sobre a segunda classe, através de um processo de amostragem.

O processo de *Oversampling* utilizando *data augmentation* gerou novos patches a partir de alterações sobre (i) forma, esticando, encolhendo e torcendo; (ii) escala, zoom; (iii) posição, rotacionando; e (iv) matiz, alteração das cores. E o processo de *Undersampling* se deu por técnicas de amostragem aleatória simples, sem repetição. As novas proporções foram de aproximadamente 1: 24 e 1: 2,5 para os *Datasets* 1 e 2, respectivamente.

Segue abaixo um resumo da condição dos datasets com as técnicas discutidas nesta Seção:

- *Dataset* 1 - 91 imagens com amastigotas e 4526 sem - 1:50;
- *Dataset* 1 Pré-processado - 91 imagens com amastigotas e 2342 sem - 1:26;
- *Dataset* 2 - 760 imagens com amastigotas e 76.495 sem - 1:101; e
- *Dataset* 2 Aumentado - 10.000 imagens com amastigotas e 25.270 sem - 1:2,5.

3.2 EXTRAÇÃO DE CARACTERÍSTICAS

Para a etapa de extração de características dos *Datasets* foram escolhidos 3 dos algoritmos mais famosos e comumente utilizados.

O primeiro foi o SIFT que funciona em 4 etapas: 1. Detecção de extremos; 2. Localização de pontos-chave; 3. Definição da orientação; e 4. Descrição dos pontos-chave. As duas primeiras

etapas descrevem a parte do detector e as duas últimas descrevem a formação do descritor (LOWE, 1999).

A primeira etapa do SIFT consiste em buscar pontos que sejam invariantes à escala da imagem. Para isto, o algoritmo utiliza uma função chamada de *scale-space*, neste caso, a função Gaussiana. Uma imagem $I(x, y)$ passa a ser definida por $L(x, y, \sigma)$ no espaço-escala, sendo esta produzida por uma convolução de uma função Gaussiana $G(x, y, \sigma)$ com a imagem $I(x, y)$, onde σ é o parâmetro de escala, ou seja, quanto maior este valor, maior o borrimento. Essa convolução é definida pela Equação (1):

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (1)$$

onde

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2/2\sigma^2)} \quad (2)$$

O algoritmo então utiliza estas imagens borradas para gerar um novo conjunto de imagens, o *Difference of Gaussians (DoG)*. O *DoG* é obtido através da diferença de borrimento Gaussiano de uma imagem com valores σ diferentes, sendo σ e $k\sigma$, onde k é uma constante de escala.

O *DoG* então é utilizado para buscar os extremos locais tanto em escala quanto em espaço, ou seja, um *pixel* é comparado com seus 8 vizinhos assim como seus 9 vizinhos da próxima escala e os 9 vizinhos da escala anterior. Se o ponto analisado for um extremo local, ele é um ponto-chave em potencial.

O segundo passo consiste em refinar os resultados do passo anterior. Como alguns dos pontos-chave detectados estão em cima de uma borda ou não possuem contraste suficiente, não são considerados pontos-chave confiáveis e precisam ser eliminados. Para alcançar esse resultado, o algoritmo utiliza uma expansão de série de Taylor do espaço-escala para obter uma localização mais precisa do extremo local e se sua intensidade nesse ponto for menor que um valor limite (conforme o artigo (LOWE, 1999), o padrão do SIFT é 0,03), ele é rejeitado.

Como o *DoG* é sensível à bordas, elas também precisam ser removidas. Isto é feito utilizando uma matriz Hessiana² de tamanho 2×2 para computar a curvatura principal. Dois autovalores³ diferentes são utilizados, um maior que o outro. Se a razão entre eles for maior que um valor limite (conforme o artigo (LOWE, 1999), o padrão do SIFT é 10) para um ponto-chave, aquele ponto-chave é descartado.

² É uma matriz quadrada ($n \times n$) das derivadas parciais de segunda ordem da função.

³ Um escalar λ diz-se um autovalor de um operador linear $A : V \rightarrow V$ se existir um vetor x diferente de zero tal que $Ax = \lambda x$.

No terceiro passo, de definição da orientação, uma vizinhança é tomada em torno da localização do ponto-chave, dependendo da escala, e a magnitude e a direção do gradiente são calculadas nessa região. Um histograma de gradientes orientados (*Histogram of Oriented Gradients* ou *HoG*) com 36 *bins* cobrindo 360 graus é criado, ele é ponderado pela magnitude do gradiente e janela circular ponderada gaussiana com σ igual a 1,5 vezes a escala do ponto-chave. O pico mais alto no histograma é obtido e qualquer pico acima de 80% dele também é considerado para calcular a orientação. Ele cria pontos-chave com a mesma localização e escala, mas direções diferentes, o que contribui para a estabilidade da correspondência.

No quarto e último passo é tomada uma vizinhança de 16x16 ao redor do ponto-chave que é dividido em 16 sub-blocos de tamanho 4x4 e para cada sub-bloco um HoG de 8 *bins* é criado, assim, um total de 128 valores *bin* estão disponíveis, sendo representado como um vetor para formar o descritor do ponto-chave. Além disso, várias medidas são tomadas para obter robustez contra mudanças de iluminação, rotação etc.

Os pontos-chave entre duas imagens são combinados identificando seus vizinhos mais próximos. Em alguns casos pode ocorrer de a segunda correspondência mais próxima estar muito perto da primeira por causa de ruído ou algum outro motivo. Em casos assim, a razão entre as duas correspondências mais próximas é tomada e, se for maior que 0,8, eles são rejeitados. Esta comparação elimina 90% das correspondências falsas enquanto descarta apenas 5% das correspondências verdadeiras (LOWE, 1999).

O segundo foi o ORB (RUBLEE et al., 2011), um algoritmo tão eficiente quanto o SIFT, porém disponibilizado gratuitamente nos pacotes Python utilizados neste estudo. O ORB se baseia nos algoritmos FAST (ROSTEN; DRUMMOND, 2006) e BRIEF (CALONDER et al., 2010) para a extração e descrição de características em imagens, respectivamente.

O FAST é caracterizado por sua eficiência computacional. Para um *pixel* que se deseja identificar como ponto-chave, o algoritmo obtém a intensidade desse pixel (I_p) juntamente com um limiar arbitrário (t). Então ele considera um círculo de 16 *pixels* ao redor do *pixel* analisado (um círculo de *Bresenham*⁴ de raio 3), checando se um número N (segundo o artigo (ROSTEN; DRUMMOND, 2006), o padrão do FAST é $N = 12$) de *pixels* contíguos estão dentro de $I_p + t$. Se sim, o *pixel* analisado é considerado um ponto-chave, senão é descartado e o algoritmo segue para o *pixel* seguinte.

Já o BRIEF converte *patches* de imagens em vetores de características binários, que são

⁴ É uma variação do *midpoint circle algorithm*, usado para determinar os pontos necessários para rasterizar um círculo.

vetores de características que contém apenas 0 e 1, para que juntos eles possam representar um objeto. No BRIEF, cada ponto-chave é descrito por um vetor de características que é uma sequência de 128 a 512 *bits*.

O ORB portanto utiliza a eficiência computacional da extração de características do FAST para extrair os pontos-chave de cada imagem da base de *patches* do pré-processamento descrito na Seção 3.1, em seguida ele utiliza a robustez do BRIEF para descrever estes pontos-chave encontrados.

O terceiro e último foi o algoritmo conhecido como Haralick Features que utiliza medidas estatísticas facilmente computadas operadas sobre matrizes de co-ocorrência de níveis de cinza (*Gray-Level Co-occurrence Matrix* ou GLCM), as quais caracterizam a textura por registrar quão frequentemente valores específicos de pixels adjacentes ocorrem (HARALICK et al., 1973).

Considerando uma imagem I de tamanho $M \times K$ e níveis de cinza no intervalo $[1, N]$, GLCM uma matriz $\mathbf{P} \in \mathbb{N}^{N \times N}$ onde cada entrada $P(i, j)$ indica o número de vezes que níveis de cinza i e j ocorrem em um dado raio e direção.

$$\mathbf{P}(i, j) = \sum_{m=1}^M \sum_{k=1}^K \begin{cases} 1, & \text{se } I(m, k) = i \text{ e } I(m + d_x, k + d_y) = j \\ 0, & \text{caso contrário} \end{cases} \quad (3)$$

Onde $\delta = (d_x, d_y)$ é o vetor de deslocamento capaz de variar o GLCM com base em diferentes direções e distâncias dos valores de nível de cinza i e j fornecidos. O próximo passo é gerar a matriz normalizada $p \in \mathbb{Q}^{N \times N}$ utilizando a Equação (4).

$$p = \frac{\mathbf{P}}{\sum_{i=1}^N \sum_{j=1}^N \mathbf{P}(i, j)} \quad (4)$$

Com a matriz p e dado que:

- $p_x(i) = \sum_{j=1}^N p(i, j)$;
- $p_y(j) = \sum_{i=1}^N p(i, j)$;
- $p_{x+y} = \sum_{i=1}^N \sum_{j=1, i+j=k}^N p(i, j), \quad k = 2, 3, \dots, 2N$;
- $p_{x-y} = \sum_{i=1}^N \sum_{j=1, |i-j|=k}^N p(i, j), \quad k = 0, 1, \dots, N-1$;
- $\mu_x = \sum_{i=1}^N i \cdot p_x(i)$;
- $\mu_y = \sum_{j=1}^N j \cdot p_y(j)$;
- $\sigma_x^2 = \sum_{i=1}^N (i - \mu_x)^2 \cdot p_x(i)$;

- $\sigma_y^2 = \sum_{j=1}^N (j - \mu_y)^2 \cdot p_y(j)$;
- $HX = - \sum_{i=1}^N p_x(i) \cdot \log p_x(i)$;
- $HY = - \sum_{i=1}^N p_y(i) \cdot \log p_y(i)$;
- $HXY = - \sum_{i=1}^N \sum_{j=1}^N p(i, j) \cdot \log p(i, j)$;
- $HXY1 = - \sum_{i=1}^N \sum_{j=1}^N p(i, j) \cdot \log (p_x(i)p_y(j))$;
- $HXY2 = - \sum_{i=1}^N \sum_{j=1}^N p_x(i) \cdot p_y(j) \cdot \log (p_x(i)p_y(j))$.

As características de textura de Haralick podem ser extraídas:

1. Segundo Momento Angular (Energia): $f_1 = \sum_{i=1}^N \sum_{j=1}^N \{p(i, j)\}^2$
2. Contraste: $f_2 = \sum_{i=1}^N \sum_{j=1}^N (i - j)^2 p(i, j)$
3. Correlação: $f_3 = \frac{\sum_{i=1}^N \sum_{j=1}^N (ij)p(i, j) - \mu_x \mu_y}{\sigma_x \sigma_y}$
4. Soma dos Quadrados (Variância): $f_4 = \sum_{i=1}^N \sum_{j=1}^N (i - \mu)^2 p(i, j)$
5. Momento Diferença Inversa: $f_5 = \sum_{i=1}^N \sum_{j=1}^N \frac{1}{1+(i-j)^2} p(i, j)$
6. Média da Soma: $f_6 = \sum_{i=2}^{2N} i p_{x+y}(i)$
7. Variância da Soma: $f_7 = \sum_{i=2}^{2N} (i - f_6)^2 p_{x+y}(i)$
8. Entropia da Soma: $f_8 = - \sum_{i=2}^{2N} p_{x+y}(i) \log (p_{x+y}(i))$
9. Entropia: $f_9 = - \sum_{i=1}^N \sum_{j=1}^N p(i, j) \log (p(i, j))$
10. Variância da Diferença: $f_{10} = \text{variance of } p_{x-y}$
11. Entropia da Diferença: $f_{11} = - \sum_{i=0}^{N-1} p_{x-y}(i) \log (p_{x-y}(i))$
12. Informação de Medidas de Correlação 1: $f_{12} = \frac{HXY - HXY1}{\max(HX, HY)}$
13. Informação de Medidas de Correlação 2: $f_{13} = \left[1 - \exp \left(- 2(HXY2 - HXY) \right) \right]^{1/2}$
14. Coeficiente de Correlação Máxima: $f_{14} = (\text{Segundo maior autovalor de } Q)^2$, onde

$$Q(i, j) = \sum_k \frac{p(i, k)p(j, k)}{p_x(i)p_y(k)}$$

3.3 CLASSIFICAÇÃO

Os algoritmos de classificação que foram utilizados foram o SVM (CORTES; VAPNIK, 1995) e o KNN (ALTMAN, 1992), ambos algoritmos de aprendizagem supervisionada usados tanto para tarefas de regressão quanto classificação.

O SVM é um modelo de AM que cria uma linha ou hiperplano⁵ para separar os dados em classes. Ele funciona representando os dados como vetores, chamados vetores de suporte, em um gráfico e projetando um hiperplano nesse gráfico de modo que haja a melhor separação possível entre os dados, ou seja, que todos os dados de uma mesma classe fiquem juntos no mesmo "lado". Para avaliar a qualidade da projeção, o SVM calcula as distâncias entre o hiperplano e os vetores de ambos os lados, essa distância é chamada de margem, e o objetivo do SVM é que essa margem seja a maior possível (CORTES; VAPNIK, 1995; PUPALE, 2018).

Já o KNN calcula a distância do dado a ser previsto com todos os dados de treino e seleciona os k pontos mais próximos do dado de teste, então ele calcula a probabilidade deste novo dado pertencer às classes baseadas nesses k pontos mais próximos, contando o número de pontos de uma mesma categoria. O dado de teste será atribuído à categoria que tiver a probabilidade mais alta (ALTMAN, 1992; CHRISTOPHER, 2021).

3.4 MÉTRICAS DE AVALIAÇÃO

As métricas de avaliação utilizadas foram Acurácia, que indica uma performance geral do modelo, Precisão, *Recall* ou Sensibilidade, F1 e Especificidade que podem ser definidas como:

- Acurácia: dentre todas as classificações, quantas o modelo classificou corretamente;
- Precisão: dentre todas as classificações de classe Positivo que o modelo fez, quantas estão corretas;
- Recall: dentre todas as situações de classe Positivo como valor esperado, quantas estão corretas;
- F1: média harmônica entre precisão e recall; e
- Especificidade: dentre todas as situações de classe Negativo como valor esperado, quantas estão corretas.

⁵ É a generalização do plano em diferentes números de dimensões.

Além destas métricas, também foi utilizado a curva ROC e a AUC. A curva ROC é um gráfico que ilustra a capacidade de diagnóstico de um sistema classificador binário à medida que seu limite de discriminação é variado. A AUC representa o grau de separabilidade entre as duas classes. Quanto maior o valor da AUC, melhor o classificador é em corretamente prever que falso é falso e verdadeiro é verdadeiro.

O Capítulo 4 apresentará resultados do uso dos métodos apresentados nas Seções 3.2 e 3.3, sobre a bases de *patches* resultantes do pré-processamento descrito na Seção 3.1.

4 RESULTADOS E DISCUSSÕES

4.1 RESULTADOS

Os experimentos de avaliação dos métodos de extração de características vistos na Seção 3.2 e classificação dos *patches* das imagens especificados na Seção 3.3 foram conduzidos separadamente nos *patches* extraídos dos *Datasets* 1 e 2.

Apesar do processo de amostragem, descrito na Seção 3.1.4, sobre a base de *patches* extremamente desbalanceada, ainda assim, o resultado desta amostragem produziu uma base com um desbalanceamento muito menor, mas ainda desbalanceada. Devido a esse fator, não foi utilizada a métrica Acurácia junto as métricas descritas na Seção 3.4. Em contextos de desbalanceamento esta métrica não descreve o comportamento real da classificação.

Isto acontece porque em bases de dados muito desbalanceadas os modelos de classificação tendem a favorecer as classes majoritárias, algumas vezes até ignorando completamente a classe minoritária, que quase sempre é a que nos interessa identificar. Por exemplo, se temos uma classe com 9900 instâncias e outra com 100, um classificador treinado com 80% dos dados usará 7920 instâncias da classe majoritária e apenas 80 da classe minoritária, portanto ele ficará muito bom em aprender as características da classe majoritária, mas não da outra, o que o torna péssimo para generalizações (QAYYUM, 2022; KOLASSA, 2017).

É importante notar que o extrator de características *Haralick* tem desvio padrão baixo e Especificidade alta pois na maioria dos testes realizados os classificadores não fizeram previsões para a classe Positiva para amastigotas de *leishmania*, exceto no *Dataset 2* Aumentado.

4.1.1 Dataset 1

Os resultados das métricas avaliadas para o classificador SVM estão presentes na Tabela 1 e os resultados do classificador KNN estão na Tabela 2. A curva ROC apenas do SVM está na Figura 6a e a do KNN está na Figura 6b e a curva com ambos os classificadores está na Figura 6c.

Tabela 1 – Dataset 1 - SVM.

	SIFT	ORB	Haralick
Acurácia	97.14 % (0.007)	94.44 % (0.005)	97.76 % (0.002)
Precisão	86.22 % (0.150)	64.57 % (0.134)	48.88 % (0.001)
Recall	64.11 % (0.038)	56.74 % (0.019)	50.00 % (0.001)
F1	69.96 % (0.059)	58.91 % (0.033)	49.43 % (0.002)
Especificidade	99.65 % (0.002)	98.48 % (0.003)	100.00 % (0.001)

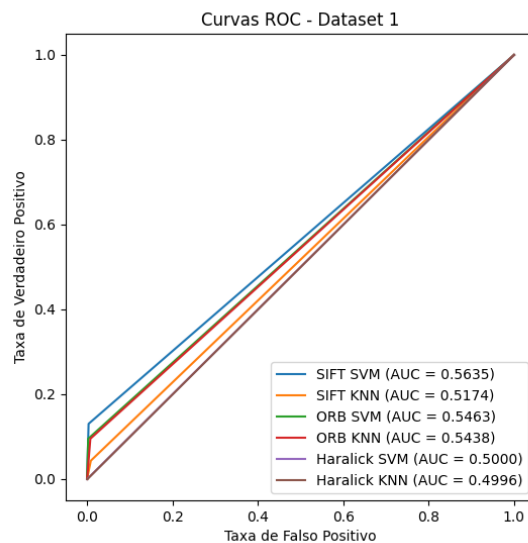
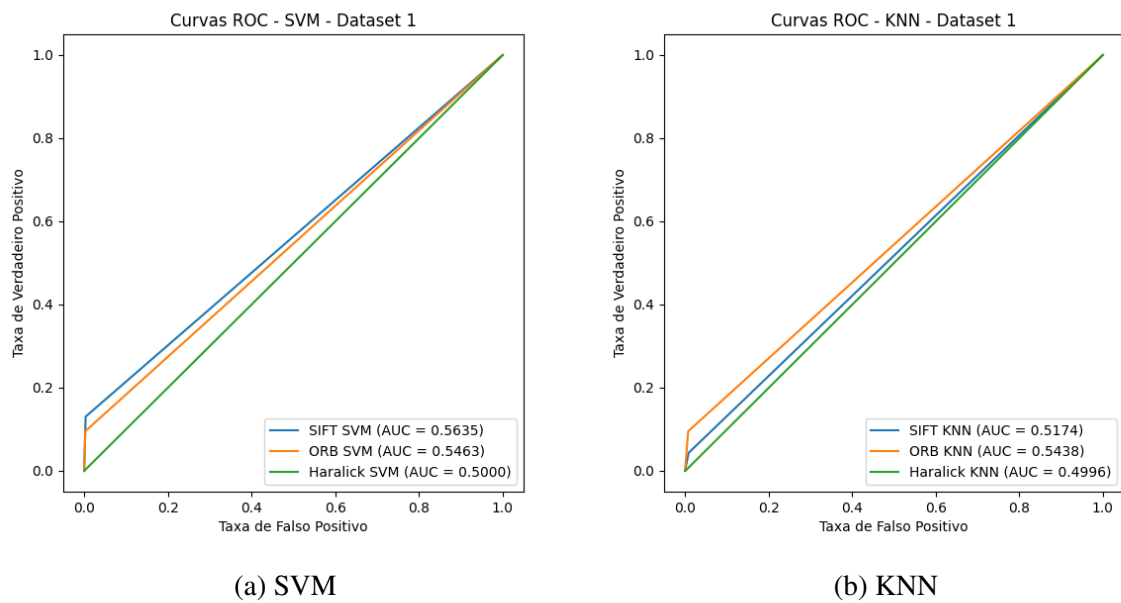
Fonte: Autor desta monografia, 2023.

Tabela 2 – Dataset 1 - KNN.

	SIFT	ORB	Haralick
Acurácia	96.63 % (0.009)	94.93 % (0.007)	97.76 % (0.002)
Precisão	81.73 % (0.186)	64.36 % (0.140)	48.88 % (0.001)
Recall	54.67 % (0.016)	52.25 % (0.028)	50.00 % (0.002)
F1	57.47 % (0.030)	53.04 % (0.048)	49.43 % (0.002)
Especificidade	99.83 % (0.003)	99.49 % (0.003)	100.00 % (0.002)

Fonte: Autor desta monografia, 2023.

Figura 6 – Curvas ROC - *Dataset 1*



Fonte: Autor desta monografia, 2023.

4.1.2 Dataset 1 Pré-processado

Os resultados das métricas avaliadas para o classificador SVM estão presentes na Tabela 3 e os resultados do classificador KNN estão na Tabela 4. A curva ROC apenas do SVM está na Figura 7a e a do KNN está na Figura 7b e a curva com ambos os classificadores está na Figura 7c.

Tabela 3 – Dataset 1 Pré-processado - SVM.

	SIFT	ORB	Haralick
Acurácia	96.38 % (0.009)	82.57 % (0.024)	95.36 % (0.007)
Precisão	73.37 % (0.228)	57.35 % (0.045)	47.68 % (0.003)
Recall	55.07 % (0.016)	56.85 % (0.028)	50.00 % (0.001)
F1	57.77 % (0.030)	57.08 % (0.032)	48.81 % (0.002)
Especificidade	99.60 % (0.002)	90.26 % (0.016)	100.00 % (0.001)

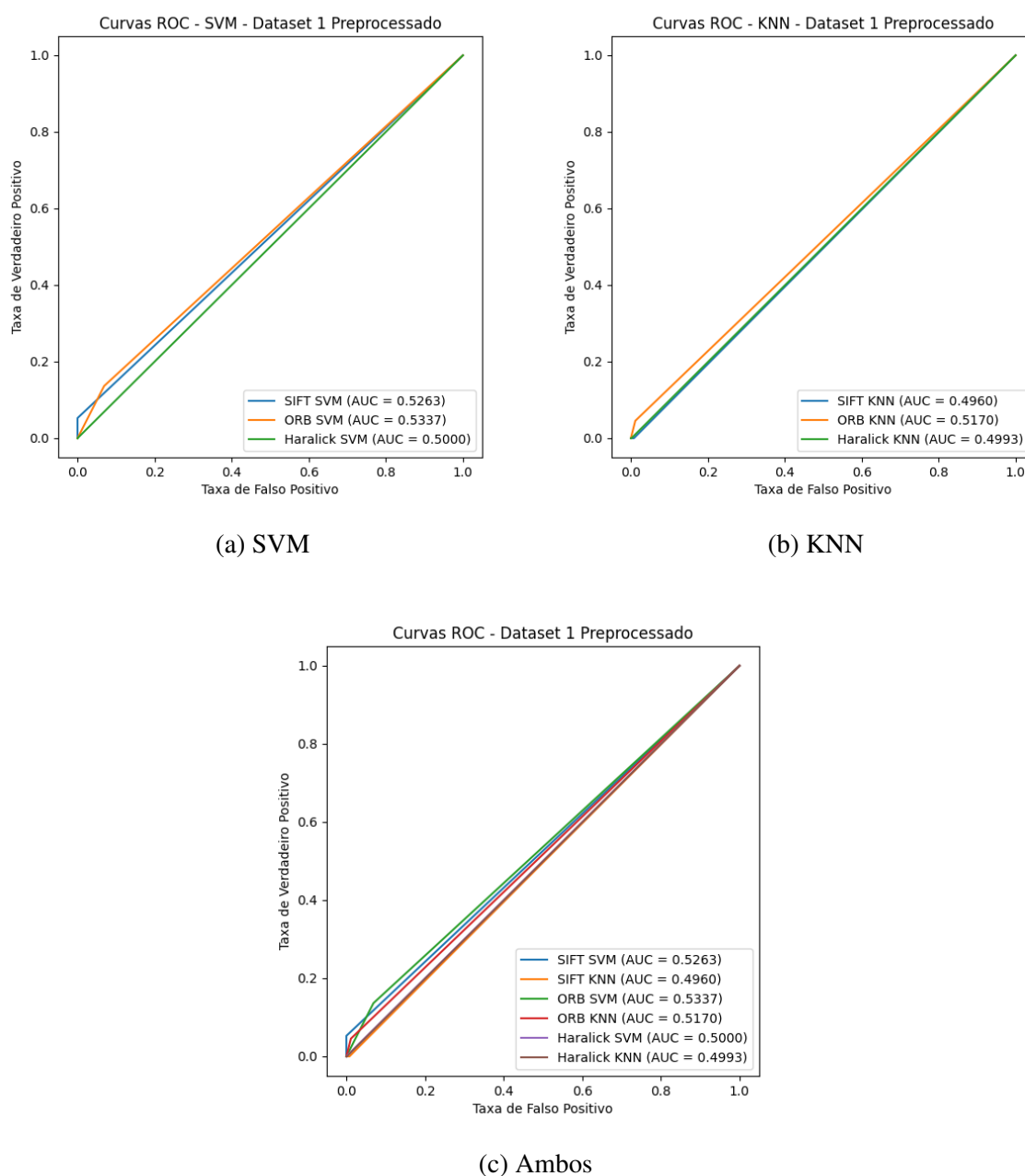
Fonte: Autor desta monografia, 2023.

Tabela 4 – Dataset 1 Pré-processado - KNN.

	SIFT	ORB	Haralick
Acurácia	96.16 % (0.005)	84.40 % (0.037)	95.36 % (0.007)
Precisão	64.96 % (0.199)	52.51 % (0.201)	47.68 % (0.003)
Recall	52.43 % (0.007)	51.24 % (0.014)	50.00 % (0.001)
F1	53.57 % (0.013)	50.99 % (0.028)	48.81 % (0.002)
Especificidade	99.60 % (0.003)	94.79 % (0.006)	100.00 % (0.001)

Fonte: Autor desta monografia, 2023.

Figura 7 – Curvas ROC - *Dataset 1* Pré-processado



Fonte: Autor desta monografia, 2023.

4.1.3 *Dataset 2*

Os resultados das métricas avaliadas para o classificador SVM estão presentes na Tabela 5 e os resultados do classificador KNN estão na Tabela 6. A curva ROC apenas do SVM está na Figura 8a e a do KNN está na Figura 8b e a curva com ambos os classificadores está na Figura 8c.

Tabela 5 – Dataset 2 - SVM.

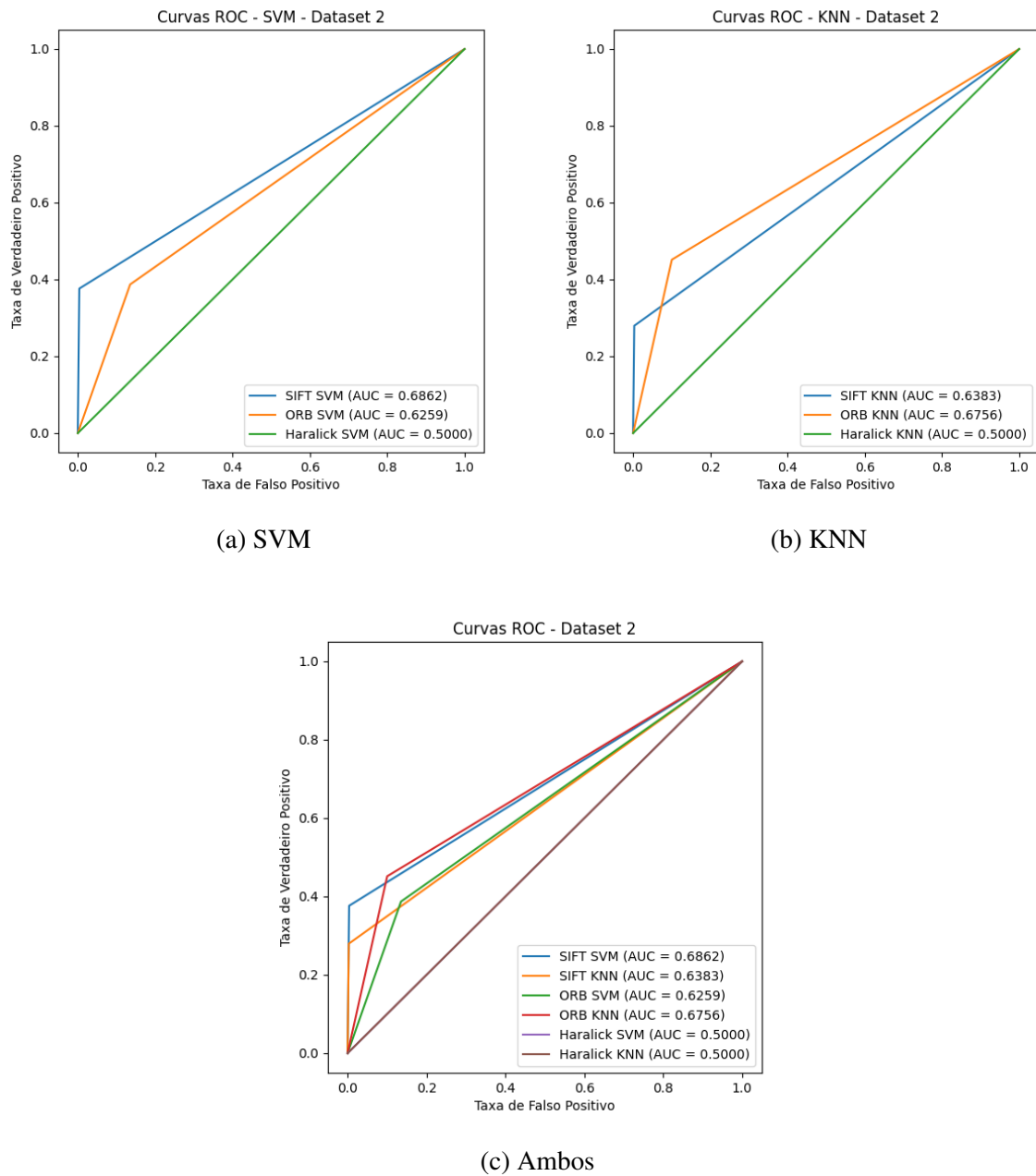
	SIFT	ORB	Haralick
Acurácia	97.90 % (0.001)	80.41 % (0.029)	99.02 % (0.002)
Precisão	84.86 % (0.013)	73.89 % (0.055)	49.51 % (0.003)
Recall	68.54 % (0.018)	67.09 % (0.046)	50.00 % (0.001)
F1	74.05 % (0.013)	69.11 % (0.050)	49.76 % (0.002)
Especificidade	99.58 % (0.001)	92.76 % (0.022)	100.00 % (0.003)

Fonte: Autor desta monografia, 2023.

Tabela 6 – Dataset 2 - KNN.

	SIFT	ORB	Haralick
Acurácia	97.77 % (0.001)	80.07 % (0.015)	99.03 % (0.002)
Precisão	82.93 % (0.031)	73.74 % (0.038)	99.51 % (0.004)
Recall	66.53 % (0.015)	65.40 % (0.041)	50.22 % (0.001)
F1	71.84 % (0.015)	67.49 % (0.040)	50.20 % (0.002)
Especificidade	99.55 % (0.001)	93.67 % (0.038)	100.00 % (0.001)

Fonte: Autor desta monografia, 2023.

Figura 8 – Curvas ROC - *Dataset 2*

Fonte: Autor desta monografia, 2023.

4.1.4 *Dataset 2* Aumentado

Os resultados das métricas avaliadas para o classificador SVM estão presentes na Tabela 7 e os resultados do classificador KNN estão na Tabela 8. A curva ROC apenas do SVM está na Figura 9a e a do KNN está na Figura 9b e a curva com ambos os classificadores está na Figura 9c.

Tabela 7 – Dataset 2 Aumentado - SVM.

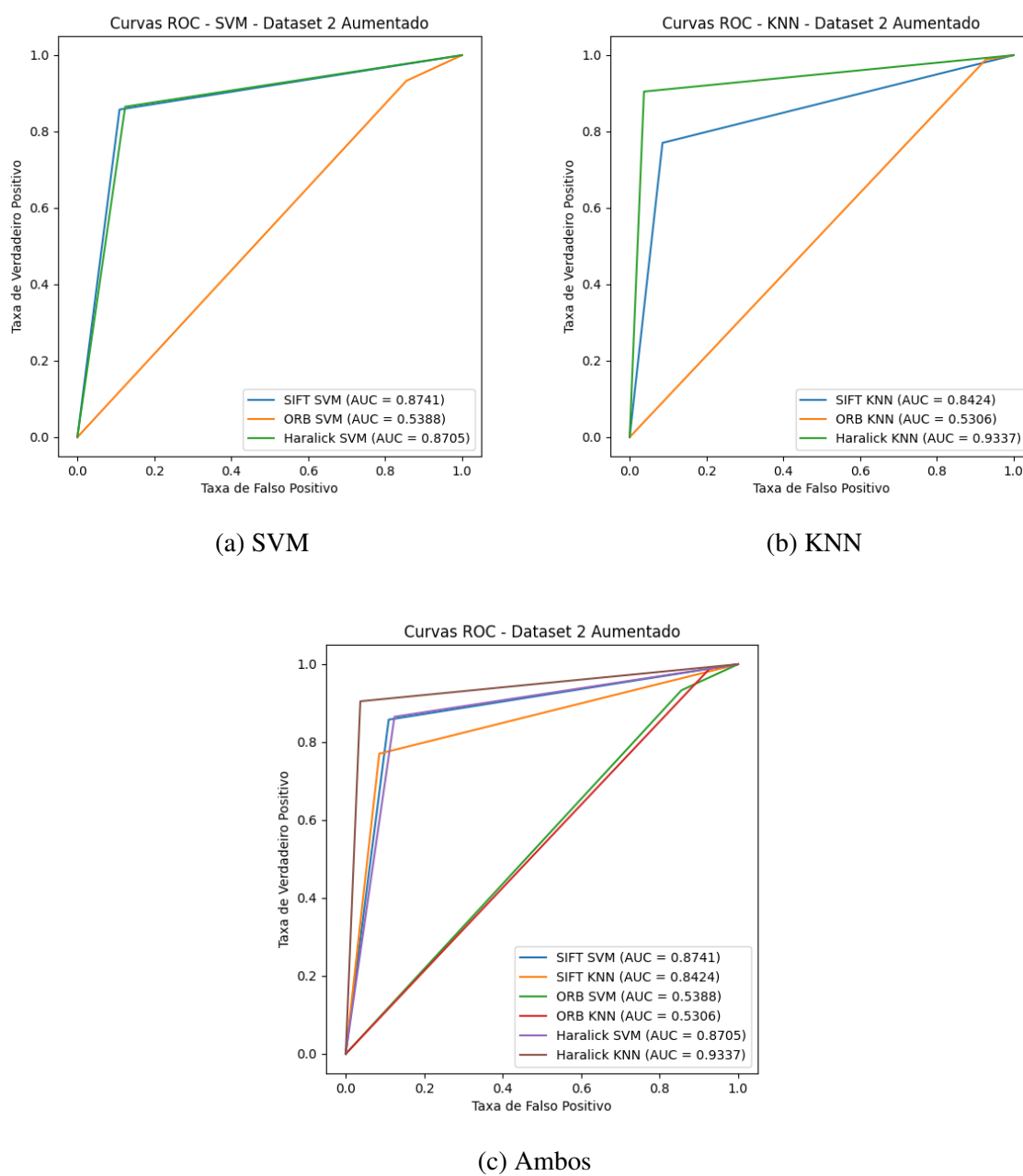
	SIFT	ORB	Haralick
Acurácia	87.37 % (0.004)	83.90 % (0.024)	87.77 % (0.004)
Precisão	87.43 % (0.004)	71.96 % (0.057)	84.18 % (0.005)
Recall	87.37 % (0.004)	60.06 % (0.028)	86.98 % (0.004)
F1	87.36 % (0.004)	62.24 % (0.037)	85.36 % (0.004)
Especificidade	89.40 % (0.006)	23.68 % (0.052)	88.75 % (0.005)

Fonte: Autor desta monografia, 2023.

Tabela 8 – Dataset 2 Aumentado - KNN.

	SIFT	ORB	Haralick
Acurácia	83.93 % (0.006)	82.54 % (0.019)	94.72 % (0.001)
Precisão	84.48 % (0.006)	63.02 % (0.115)	93.56 % (0.002)
Recall	83.94 % (0.006)	51.43 % (0.037)	93.18 % (0.002)
F1	83.87 % (0.011)	48.80 % (0.054)	93.37 % (0.002)
Especificidade	90.28 % (0.007)	3.95 % (0.097)	96.62 % (0.001)

Fonte: Autor desta monografia, 2023.

Figura 9 – Curvas ROC - *Dataset 2* Aumentado

Fonte: Autor desta monografia, 2023.

4.1.5 SIFT

As Tabelas 9 à 12 são referentes apenas ao algoritmo SIFT, mostrando os resultados por *Dataset* e dos dois classificadores usados.

Tabela 9 – SIFT - Dataset 1.

	SVM	KNN
Acurácia	97.14 % (0.007)	96.63 % (0.009)
Precisão	86.22 % (0.150)	81.73 % (0.186)
Recall	64.11 % (0.038)	54.67 % (0.016)
F1	69.96 % (0.059)	57.47 % (0.030)
Especificidade	99.65 % (0.002)	99.83 % (0.003)

Fonte: Autor desta monografia, 2023.

Tabela 10 – SIFT - Dataset 1 Pré-processado.

	SVM	KNN
Acurácia	96.38 % (0.009)	96.16 % (0.005)
Precisão	73.37 % (0.228)	64.96 % (0.199)
Recall	55.07 % (0.016)	52.43 % (0.007)
F1	57.77 % (0.030)	53.57 % (0.013)
Especificidade	99.60 % (0.002)	99.60 % (0.003)

Fonte: Autor desta monografia, 2023.

Tabela 11 – SIFT - Dataset 2.

	SVM	KNN
Acurácia	97.90 % (0.001)	97.77 % (0.001)
Precisão	84.86 % (0.013)	82.93 % (0.031)
Recall	68.54 % (0.018)	66.53 % (0.015)
F1	74.05 % (0.013)	71.84 % (0.015)
Especificidade	99.58 % (0.001)	99.55 % (0.001)

Fonte: Autor desta monografia, 2023.

Tabela 12 – SIFT - Dataset 2 Aumentado.

	SVM	KNN
Acurácia	87.37 % (0.004)	83.93 % (0.006)
Precisão	87.43 % (0.004)	84.48 % (0.006)
Recall	87.37 % (0.004)	83.94 % (0.006)
F1	87.36 % (0.004)	83.87 % (0.011)
Especificidade	89.40 % (0.006)	90.28 % (0.007)

Fonte: Autor desta monografia, 2023.

4.1.6 ORB

As Tabelas 13 à 16 são referentes apenas ao algoritmo ORB, cada tabela é referente à um *Dataset* e aos dois classificadores usados.

Tabela 13 – ORB - Dataset 1.

	SVM	KNN
Acurácia	94.44 % (0.005)	94.93 % (0.007)
Precisão	64.57 % (0.134)	64.36 % (0.140)
Recall	56.74 % (0.019)	52.25 % (0.028)
F1	58.91 % (0.033)	53.04 % (0.048)
Especificidade	98.48 % (0.003)	99.49 % (0.003)

Fonte: Autor desta monografia, 2023.

Tabela 14 – ORB - Dataset 1 Pré-processado.

	SVM	KNN
Acurácia	82.57 % (0.024)	84.40 % (0.037)
Precisão	57.35 % (0.045)	52.51 % (0.201)
Recall	56.85 % (0.028)	51.24 % (0.014)
F1	57.08 % (0.032)	50.99 % (0.028)
Especificidade	90.26 % (0.016)	94.79 % (0.006)

Fonte: Autor desta monografia, 2023.

Tabela 15 – ORB - Dataset 2.

	SVM	KNN
Acurácia	80.41 % (0.029)	80.07 % (0.015)
Precisão	73.89 % (0.055)	73.74 % (0.038)
Recall	67.09 % (0.046)	65.40 % (0.041)
F1	69.11 % (0.050)	67.49 % (0.040)
Especificidade	92.76 % (0.022)	93.67 % (0.038)

Fonte: Autor desta monografia, 2023.

Tabela 16 – ORB - Dataset 2 Aumentado.

	SVM	KNN
Acurácia	83.90 % (0.024)	82.54 % (0.019)
Precisão	71.96 % (0.057)	63.02 % (0.115)
Recall	60.06 % (0.028)	51.43 % (0.037)
F1	62.24 % (0.037)	48.80 % (0.054)
Especificidade	23.68 % (0.052)	3.95 % (0.097)

Fonte: Autor desta monografia, 2023.

4.1.7 Haralick

As Tabelas 17, 18, 19 e 20 são referentes apenas ao algoritmo Haralick, cada tabela é referente à um *Dataset* e aos dois classificadores usados.

Tabela 17 – Haralick - Dataset 1.

	SVM	KNN
Acurácia	97.76 % (0.002)	97.76 % (0.002)
Precisão	48.88 % (0.001)	48.88 % (0.001)
Recall	50.00 % (0.001)	50.00 % (0.002)
F1	49.43 % (0.002)	49.43 % (0.002)
Especificidade	100.00 % (0.001)	100.00 % (0.002)

Fonte: Autor desta monografia, 2023.

Tabela 18 – Haralick - Dataset 1 Pré-processado.

	SVM	KNN
Acurácia	95.36 % (0.007)	95.36 % (0.007)
Precisão	47.68 % (0.003)	47.68 % (0.003)
Recall	50.00 % (0.001)	50.00 % (0.001)
F1	48.81 % (0.002)	48.81 % (0.002)
Especificidade	100.00 % (0.001)	100.00 % (0.001)

Fonte: Autor desta monografia, 2023.

Tabela 19 – Haralick - Dataset 2.

	SVM	KNN
Acurácia	99.02 % (0.002)	99.03 % (0.002)
Precisão	47.68 % (0.003)	99.51 % (0.004)
Recall	50.00 % (0.001)	50.22 % (0.001)
F1	48.81 % (0.002)	50.22 % (0.002)
Especificidade	100.00 % (0.003)	100.00 % (0.001)

Fonte: Autor desta monografia, 2023.

Tabela 20 – Haralick - Dataset 2 Aumentado.

	SVM	KNN
Acurácia	87.77 % (0.004)	94.72 % (0.001)
Precisão	84.18 % (0.005)	93.56 % (0.002)
Recall	86.98 % (0.004)	93.18 % (0.002)
F1	85.36 % (0.004)	93.37 % (0.002)
Especificidade	88.75 % (0.005)	96.62 % (0.001)

Fonte: Autor desta monografia, 2023.

4.2 DISCUSSÃO

Como é possível perceber em praticamente todas as tabelas de resultados, o algoritmo SIFT se saiu consideravelmente melhor que os outros, com o algoritmo ORB em segundo lugar e por último o Haralick, o que se manteve de acordo com outras instâncias de comparações entre o SIFT e outros extratores de características, com o SIFT sempre apresentando uma robustez

grande e maior quantidade de pontos-chave (BANSAL et al., 2021). A única exceção foi nos resultados do *Dataset 2 Aumentado* do classificador KNN, como podemos ver na Tabela 8, onde o Haralick foi quem se saiu melhor.

Observando as tabelas referentes ao algoritmo SIFT, as Tabelas 9 à 12, podemos perceber que independentemente de qual *Dataset* seja usado, o classificador SVM é o que sempre apresenta os melhores resultados. Na literatura este resultado, de SIFT + SVM obter resultados melhores do que SIFT + KNN é comum, mas as diferenças entre eles normalmente são pequenas (AL-SHEHRI et al., 2017; SIDANI et al., 2019; SHAIENDRA; ANITHA, 2020).

Quando comparamos os resultados apenas do algoritmo ORB, nas Tabelas 13 à 16, podemos perceber que o mesmo acontece, o classificador SVM se sobressai, mesmo que em alguns *Datasets* seja com pouca margem, quando comparado com o classificador KNN. O algoritmo ORB é uma alternativa eficiente para o SIFT, mas que não consegue alcançar os mesmo resultados, ficando sempre um pouco aquém dos resultados do SIFT (BANSAL et al., 2021; KUMAR et al., 2022).

E finalmente com os resultados do Haralick podemos notar que no *Dataset 1* (Tabela 17) e no *Dataset 1 Pré-processado* (Tabela 18) o algoritmo teve resultados iguais. Isto se deve ao fato de que nestes *Datasets* quase nenhum dos testes e treinos dos classificadores gerou resultados em que houvessem classificações positivas para amastigotas de *leishmania*. Foram alteradas as proporções de *Dataset* para treino e teste e parâmetros dos classificadores mas o resultado se manteve o mesmo. O algoritmo Haralick comumente apresenta bons resultados, principalmente em imagens onde as texturas são mais marcantes, mas mesmo assim dificilmente consegue obter resultados melhores que SIFT ou ORB (RUBERTO; FODDE, 2014; ROY et al., 2021).

Entretanto, no *Dataset 2* (Tabela 19) e *Dataset 2 Aumentado* (Tabela 20) o padrão que vinha se desenhando nos dois outros algoritmos, do classificador SVM apresentar resultados melhores, foi quebrado, de modo que em ambos os *Datasets* o classificador KNN apresentou os melhores resultados, sendo neste último, melhores até mesmo que os resultados do algoritmo SIFT para o mesmo *Dataset*.

Agora observando especificamente os resultados entre os *Datasets* originais e os resultantes do pré-processamento descrito na Seção 3.1 podemos observar que no *Dataset 1*, em todos os 3 extratores de características, o resultado foi melhor no *Dataset* sem o pré-processamento como é possível ver nas Tabelas 9 e 10 para o algoritmo SIFT, nas Tabelas 13 e 14 para o algoritmo ORB e nas Tabelas 17 18 para o Haralick.

Quanto ao *Dataset 2* os resultados foram mais variados, onde o SIFT e o Haralick apre-

sentaram um resultado melhor no *Dataset 2* Aumentado, com este sendo melhor no classificador KNN enquanto aquele foi melhor no classificador SVM, como é possível ver nas Tabelas 11 e 12 e Tabelas 19 e 20, respectivamente. E o ORB apresentou melhor resultado no *Dataset 2* sem pré-processamento, como pode-se observar nas Tabelas 15 e 16.

Podemos concluir disto que da amostragem de dados, explicada na Seção 3.1.4, apenas a realização do *downsampling* não é suficiente para melhorar a qualidade dos resultados, sendo necessária também a realização do *oversampling*.

Fica evidente que o algoritmo SIFT, juntamente com o classificador SVM, é a melhor opção dentre estas avaliadas neste trabalho, especialmente quando se trata de um problema com um desbalanceamento de dados enorme como este onde há uma diferença enorme na quantidade de dados para cada classe, tornando uma boa extração de características da classe minoritária imprescindível para que obtenham-se resultados aceitáveis e consistentes, de modo que possam ser aplicados no mundo real (BANSAL et al., 2021).

O Capítulo 5 apresentará a conclusão deste trabalho, com um apanhado geral do que foi discutido neste capítulo, e também discorrerá sobre possíveis trabalhos futuros.

5 CONCLUSÃO

Neste trabalho avaliamos três dos principais algoritmos de extração de características atualmente, obtendo resultados que se mantiveram semelhantes em praticamente todos os testes, com o algoritmo SIFT sendo o que apresentou melhor resultado especialmente ao ser combinado com o classificador SVM, onde todas as métricas avaliadas foram superiores aos testes com o mesmo algoritmo mas com o classificador KNN em todos os *Datasets* avaliados.

Fica bastante evidente que o algoritmo SIFT é a melhor opção dentre estas avaliadas neste trabalho, onde é necessária uma boa extração de características da classe minoritária, pelo enorme desbalanceamento na quantidade de dados de cada classe, para que obtenha-se resultados aceitáveis e confiáveis, de modo que possam ser utilizados no mundo real.

Também é possível notar que a amostragem de dados descrita na Seção 3.1.4 não trouxe um aumento nas métricas avaliadas quando realizado apenas o *undersampling*, utilizado no *Dataset 1*, fazendo-se necessário o uso tanto do *undersampling* quanto do *oversampling*, utilizado no *Dataset 2*.

Como trabalhos futuros, é interessante aumentar o escopo do trabalho como um todo, com mais extratores de características e classificadores, incluindo também redes neurais, tanto modelos mais simples como um Perceptron quanto redes mais complexas como modelos de Aprendizado Profundo.

REFERÊNCIAS

Adobe Inc. **Adobe Photoshop**. 2023. Disponível em: <<https://www.adobe.com/products/photoshop.html>>.

AL-SHEHRI, H.; AL-QARNI, A.; AL-SAATI, L.; BATOAQ, A.; BADUKHEN, H.; ALRASHED, S.; ALHIYAFI, J.; OLATUNJI, S. O. Student performance prediction using support vector machine and k-nearest neighbor. In: IEEE. **2017 IEEE 30TH CANADIAN CONFERENCE ON ELECTRICAL AND COMPUTER ENGINEERING (CCECE)**. [S.l.], 2017. (Canadian Conference on Electrical and Computer Engineering). ISBN 978-1-5090-5538-8. ISSN 0840-7789. 30th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), Windsor, CANADA, APR 30-MAY 03, 2017.

ALTMAN, N. S. An introduction to kernel and nearest-neighbor nonparametric regression. **The American Statistician**, Taylor & Francis Group, v. 46, n. 3, p. 175–185, 1992.

ALVES, W. A. Leishmaniose visceral americana: situação atual no brasil. **BEPA. Boletim Epidemiológico Paulista**, v. 6, n. 71, p. 25–29, 2009.

BANSAL, M.; KUMAR, M.; KUMAR, M. 2d object recognition: a comparative analysis of sift, surf and orb feature descriptors. **Multimedia Tools and Applications**, v. 80, n. 12, p. 18839–18857, May 2021. ISSN 1573-7721. Disponível em: <<https://doi.org/10.1007/s11042-021-10646-0>>.

BLEI, D. M.; TALWALKAR, A.; JAIN, A. Improving the accuracy of medical diagnosis with causal machine learning. **Nature communications**, Nature Publishing Group, v. 11, n. 1, p. 1–14, 2020.

BRASIL. Ministério da Saúde. **Manual de Vigilância e Controle da Leishmaniose Visceral**. 1. ed. [S.l.]: Editora MS, 2006.

BRASIL. Ministério da Saúde, **Ciclo de Transmissão da Leishmaniose Visceral**. 2019. Acesso em: 21 de maio de 2023. Disponível em: <<https://www.saude.ba.gov.br/temasdesaude/transmissao-leishmaniose-visceral/>>.

BRASIL. Ministério da Saúde, **Leishmaniose Visceral**. 2022. Acesso em: 10 de maio de 2023. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/saude-de-a-a-z/l/leishmaniose-visceral>>.

BRASIL. Ministério da Saúde, **Situação epidemiológica da Leishmaniose Visceral**. 2022. Acesso em: 10 de maio de 2023. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/saude-de-a-a-z/l/leishmaniose-visceral/situacao-epidemiologica-da-leishmaniose-visceral>>.

BURKOV, A. **Machine learning in action**. [S.l.]: Manning Publications Co., 2019.

BUSH, S. G.; WATSON, J. G.; LAAN, M. A. J. van der. A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: a systematic review and meta-analysis. **Science Direct**, Elsevier, 2020.

CALONDER, M.; LEPETIT, V.; STRECHA, C.; FUA, P. Brief: Binary robust independent elementary features. In: DANIILIDIS, K.; MARAGOS, P.; PARAGIOS, N. (Ed.). **Computer Vision – ECCV 2010**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 778–792. ISBN 978-3-642-15561-1.

- CARNEIRO, M.; MORENO, E. C.; GONÇALVES, A. V.; LAMBERTUCCI, J. R.; ANTUNES, C. M. Visceral leishmaniasis: Challenges in identifying subclinical leishmania infection. **Drug Development Research**, v. 72, n. 6, p. 442–450, 2011. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/ddr.20451>>.
- CHEN, L.; DOU, Q.; HENG, P.-A.; WANG, K.; XU, K.; KHOR, S.-H.; HENG, S.-L. Deep learning for medical image analysis: A survey. **IEEE Access**, IEEE, v. 8, p. 103051–103070, 2020.
- CHRISTOPHER, A. **K-Nearest Neighbors**. 2021. Acesso em: 24 de maio de 2023. Disponível em: <<https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4>>.
- COELHO, L. P. Mahotas: Open source software for scriptable computer vision. **Journal of Open Research Software**, Jul 2013.
- CORTES, C.; VAPNIK, V. Support-vector networks. **Machine learning**, Springer, v. 20, n. 3, p. 273–297, 1995.
- CUI, S.; JIANG, H.; WANG, Z.; SHEN, C. Application of neural network based on sift local feature extraction in medical image classification. In: IEEE; SICHUAN PROV COMP SCI; SINGAPORE INST ELECT; CHENGDU UNIV INFORMAT TECHNOL; CHINESE ACAD SCI CO LTD, CHENGDU INFORMAT TECHNOL. **2017 2ND INTERNATIONAL CONFERENCE ON IMAGE, VISION AND COMPUTING (ICIVC 2017)**. [S.l.], 2017. p. 92–97. ISBN 978-1-5090-6238-6. 2nd International Conference on Image, Vision and Computing (ICIVC), Chengdu, PEOPLES R CHINA, JUN 02-04, 2017.
- DOUGHERTY, E. R. **Image processing in medical applications**. [S.l.]: Academic Press, 2014.
- ELMAHALLAWY, E. K.; SAMPEDRO, A. M.; RODRIGUEZ-GRANGER, J.; HOYOS-MALLECOT, Y.; AGIL, A.; NAVARRO, J. M.; FERNÁNDEZ, J. Diagnosis of leishmaniasis. **Journal of Infection in Developing Countries**, v. 8, n. 8, p. 961 – 972, 2014.
- ESTEVA, A.; KUPREL, B.; NOVOA, J. M.; KO, J.; SWETTER, S. M.; BLAU, A. D.; THRUN, S. Dermatologist-level classification of skin cancer with deep neural networks. **Nature Medicine**, Nature Publishing Group, v. 23, n. 11, p. 1596–1603, 2017.
- FARAH, M.; RABBANI, H.; TALEBI, A. Automatic boundary extraction of leishman bodies in bone marrow samples from patients with visceral leishmaniasis. **Journal of Isfahan Medical School**, v. 32, n. 286, Jul 2014.
- FUENTE, J. M. De la; ARENAS, R. Diagnosis of leishmaniasis. **Clinical Microbiology Reviews**, American Society for Microbiology, v. 21, n. 4, p. 749–777, 2008.
- GHOSH, A.; SENGUPTA, S.; BANDYOPADHYAY, S.; PAL, A.; ROY, J.; MAJUMDER, D. Machine learning-based diagnosis of skin diseases: a systematic review and meta-analysis. **Journal of the American Medical Informatics Association**, v. 27, n. 1, p. 107–115, 2020.
- GONZALEZ, R. C.; WOODS, R. E. **Digital image processing**. [S.l.]: Prentice Hall, 2002.
- GULATI, S.; AGRAWAL, A.; MITTAL, A.; GARG, A. Automated detection of diabetic retinopathy using deep learning. **Nature Medicine**, Nature Publishing Group, v. 23, n. 11, p. 1413–1418, 2017.

- HARALICK, R. M.; SHANMUGAM, K.; DINSTEN, I. Textural features for image classification. **IEEE Transactions on Systems, Man, and Cybernetics**, SMC-3, n. 6, p. 610–621, 1973.
- HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **Elements of statistical learning**. [S.l.]: Springer, 2009.
- HERRERO, M.; MORENO, F. Diagnosis of leishmaniasis. In: _____. **Leishmaniasis: From parasite to patient**. [S.l.]: Springer, 2010. p. 105–126.
- HUNTER, J. D. Matplotlib: A 2d graphics environment. **Computing in Science & Engineering**, IEEE COMPUTER SOC, v. 9, n. 3, p. 90–95, 2007.
- ITSEEZ. **Open Source Computer Vision Library**. 2015. <<https://github.com/itseez/opencv>>.
- KAUFFMAN, A. **Image processing for computer vision**. [S.l.]: Morgan Kaufmann, 2007.
- KHORASANI, A.; POURESLAMI, E.; ASADI, M.; SADEGHI, A.; AKBARI, A.; EBRAHIMI, A. Machine learning for predicting cardiovascular risk from electronic health records: a systematic review and meta-analysis. **Journal of medical Internet research**, v. 21, n. 11, p. e17643, 2019.
- KOLASSA, S. **Why is accuracy not the best measure for assessing classification models?** 2017. Cross Validated. Acesso em: 25 de maio de 2023. Disponível em: <<https://stats.stackexchange.com/q/312787>>.
- KUMAR, A.; PANDEY, S. C.; SAMANT, M. A spotlight on the diagnostic methods of a fatal disease visceral leishmaniasis. **Parasite Immunology**, v. 42, n. 10, p. e12727, 2020. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1111/pim.12727>>.
- KUMAR, S.; JANET, B.; NEELAKANTAN, S. Identification of malware families using stacking of textural features and machine learning. **EXPERT SYSTEMS WITH APPLICATIONS**, v. 208, DEC 1 2022. ISSN 0957-4174.
- LEE, J.-H.; SEO, Y.-H.; KIM, K.-H.; KIM, J.-W.; KANG, K.-H.; KIM, D.-H. Skin cancer image analysis with deep learning: A comprehensive review. **IEEE Access**, IEEE, v. 7, p. 106325–106340, 2019.
- LEIBOWITZ, D. Ai now diagnoses disease better than your doctor, study finds. **Towards Data Science**, Towards Data Science, 2020.
- LOWE, D. G. Object recognition from local scale-invariant features. In: IEEE. **Proceedings of the seventh IEEE international conference on computer vision**. [S.l.], 1999. v. 2, p. 1150–1157.
- MARINHO, T. T. **Aspectos Clínicos Laboratoriais e o uso da Modelagem Computacional no diagnóstico da Leishmaniose Visceral**. Dissertação (Mestrado) — Universidade Federal de Alagoas - UFAL, 2020.
- MONDAL, P.; MUKHOPADHYAY, J.; SURAL, S.; BHAAACHARYYA, P. P. 3d-sift feature based brain atlas generation: An application to early diagnosis of alzheimer's disease. In: G L BAJAJ INST TECHNOL & MANAGEMENT; IEEE U P SECT; IEEE EMB BANGALORE CHAPTER; INST COMMUN ENGINEERS & INFORMAT TECHNOLOGISTS. **2014 INTERNATIONAL CONFERENCE ON MEDICAL IMAGING, M-HEALTH &**

EMERGING COMMUNICATION SYSTEMS (MEDCOM). [S.l.], 2015. p. 342–347. ISBN 978-1-4799-5097-3. International Conference on Medical Imaging, M-Health and Emerging Communication Systems (MedCom), Greater Noida, INDIA, NOV 07-08, 2014.

MURPHY, K. P. **Machine learning: a probabilistic perspective.** [S.l.]: The MIT Press, 2012.

OMS. Organização Mundial de Saúde, **Leishmaniasis.** World Health Organization, 2020. Acesso em: 10 de maio de 2023. Disponível em: <<https://www.who.int/data/gho/data/themes/topics/topic-details/GHO/leishmaniasis>>.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V. et al. Scikit-learn: Machine learning in python. **the Journal of machine Learning research**, JMLR. org, v. 12, p. 2825–2830, 2011.

PUPALE, R. **Support Vector Machines(SVM) — An Overview.** 2018. Acesso em: 24 de maio de 2023. Disponível em: <<https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989>>.

QAYYUM, R. **Why Accuracy Is Not A Good Metric For Imbalanced Data.** 2022. Acesso em: 25 de maio de 2023. Disponível em: <<https://towardsai.net/p/why-accuracy-is-not-a-good-metric-for-imbalanced-data>>.

RAJKOMAR, A.; DEAN, J.; KOHANE, I. Machine learning in healthcare: reducing preventable deaths from heart disease. **Nature medicine**, v. 25, n. 11, p. 1880–1887, 2019.

RAJKOMAR, A.; MONGA, V.; CHEN, K.; DEHGHANI, M.; HARDT, M.; TALWALKAR, A.; SUN, J. Scalable and accurate deep learning with electronic health records. **Nature Medicine**, Nature Publishing Group, v. 25, n. 11, p. 1875–1880, 2019.

ROSTEN, E.; DRUMMOND, T. Machine learning for high-speed corner detection. In: . [S.l.: s.n.], 2006. v. 3951. ISBN 978-3-540-33832-1.

ROY, S. D.; DAS, S.; KAR, D.; SCHWENKER, F.; SARKAR, R. Computer aided breast cancer detection using ensembling of texture and statistical image features. **Sensors**, v. 21, n. 11, 2021. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/21/11/3628>>.

RUBERTO, C. D.; FODDE, G. Statistical features for image retrieval a quantitative comparison. In: **2014 International Conference on Computer Vision Theory and Applications (VISAPP).** [S.l.: s.n.], 2014. v. 1, p. 610–617.

RUBLEE, E.; RABAUD, V.; KONOLIGE, K.; BRADSKI, G. Orb: An efficient alternative to sift or surf. In: **2011 International Conference on Computer Vision.** [S.l.: s.n.], 2011. p. 2564–2571.

SHAILENDRA, R.; ANITHA, J. Enrich ayurveda knowledge using machine learning techniques. **Indian Journal of Traditional Knowledge**, v. 19, p. 813–820, 12 2020.

SHIN, D.-H.; KIM, J.-W.; KANG, K.-H.; SEO, Y.-H.; KIM, K.-H.; LEE, J.-H. Deep learning for dermoscopic image analysis: A systematic review. **IEEE Access**, IEEE, v. 6, p. 50962–50976, 2018.

SIDANI, A. A.; CHERRY, A.; HAJJ-HASSAN, H.; HAJJ-HASSAN, M. Comparison between k-nearest neighbor and support vector machine algorithms for ppg biometric identification. In: **2019 Fifth International Conference on Advances in Biomedical Engineering (ICABME)**. [S.l.: s.n.], 2019. p. 1–4.

SINGH, A.; KUMAR, A.; AGGARWAL, A.; YADAV, A.; GUPTA, A.; KUMAR, A.; SINGH, A.; KUMAR, A.; KUMAR, A.; KUMAR, A. Deep learning for the diagnosis of visceral leishmaniasis: A systematic review. **Journal of medical virology**, Wiley, v. 90, n. 11, p. 2493–2502, 2018.

SINGH, L. K.; POOJA; GARG, H.; KHANNA, M. An iot based predictive modeling for glaucoma detection in optical coherence tomography images using hybrid genetic algorithm. **MULTIMEDIA TOOLS AND APPLICATIONS**, v. 81, n. 26, p. 37203–37242, NOV 2022. ISSN 1380-7501.

TAHIR, M.; BOURIDANE, A.; KURUGOLLU, F. An fpga based coprocessor for glcm and haralick texture features and their application in prostate cancer classification. **ANALOG INTEGRATED CIRCUITS AND SIGNAL PROCESSING**, v. 43, n. 2, p. 205–215, MAY 2005. ISSN 0925-1030. 10th IEEE International Conference on Electronics, Circuits and Systems, Sharjah, U ARAB EMIRATES, DEC 14-17, 2003.

WU, H.; XU, R.; XU, K.; ZHAO, J.; ZHANG, Y.; WANG, A.; IWAHORI, Y. 3d texture reconstruction of abdominal cavity based on monocular vision slam for minimally invasive surgery. **SYMMETRY-BASEL**, v. 14, n. 2, FEB 2022.

ZHANG, J.; CHEN, X.; LIU, X.; ZHANG, Y.; LI, J.; WANG, X. Machine learning for the diagnosis of diabetes mellitus: a systematic review and meta-analysis. **Journal of medical Internet research**, v. 22, n. 12, p. e26816, 2020.