



**UNIVERSIDADE FEDERAL DA FRONTEIRA SUL
CAMPUS DE CHAPECÓ
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

RODOLFO TREVISOL

**CLASSIFICAÇÃO DE CÉLULAS BRANCAS A PARTIR DE AMOSTRAS DE
SANGUE USANDO REDE NEURAL ARTIFICIAL CONVOLUCIONAL**

**CHAPECÓ
2023**

RODOLFO TREVISOL

**CLASSIFICAÇÃO DE CÉLULAS BRANCAS A PARTIR DE AMOSTRAS DE
SANGUE USANDO REDE NEURAL ARTIFICIAL CONVOLUCIONAL**

Trabalho de conclusão de curso apresentado como requisito para obtenção do grau de Bacharel em Ciência da Computação da Universidade Federal da Fronteira Sul.
Orientador: Prof. Dr. Samuel da Silva Feitosa

CHAPECÓ
2023

Bibliotecas da Universidade Federal da Fronteira Sul - UFFS

Trevisol, Rodolfo
CLASSIFICAÇÃO DE CÉLULAS BRANCAS A PARTIR DE AMOSTRAS
DE SANGUE USANDO REDE NEURAL ARTIFICIAL CONVOLUCIONAL /
Rodolfo Trevisol. -- 2023.
59 f.:il.

Orientador: Doutor Samuel da Silva Feitosa

Trabalho de Conclusão de Curso (Graduação) -
Universidade Federal da Fronteira Sul, Curso de
Bacharelado em Ciência da Computação, Chapecó, SC, 2023.

1. Redes Neurais Convolucionais. 2. Hemograma. 3.
Classificação. 4. Células Brancas. I. Feitosa, Samuel da
Silva, orient. II. Universidade Federal da Fronteira
Sul. III. Título.

RODOLFO TREVISOL

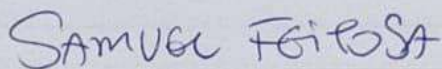
**CLASSIFICAÇÃO DE CÉLULAS BRANCAS A PARTIR DE AMOSTRAS DE
SANGUE USANDO REDE NEURAL ARTIFICIAL CONVOLUCIONAL**

Trabalho de conclusão de curso apresentado como requisito para obtenção do grau de Bacharel em Ciência da Computação da Universidade Federal da Fronteira Sul.

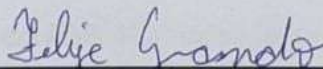
Orientador: Prof. Dr. Samuel da Silva Feitosa

Este trabalho de conclusão de curso foi defendido e aprovado pela banca avaliadora em: 14/2/2023.

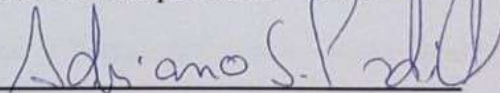
BANCA AVALIADORA



Prof. Dr. Samuel da Silva Feitosa – UFFS



Prof. Dr. Felipe Grando – UFFS



Prof. Me. Adriano Sanick Padilha – UFFS

AGRADECIMENTOS

Quero agradecer a minha mãe que não mediu, e não mede, esforços e se sacrifica todos os dias para que eu chegue até aqui. Agradecer a todo o apoio durante esta etapa da minha vida, a todas as conversas para manter a sanidade, todos os puxões de orelha e lições. Minha irmã, que deposito toda minha esperança e amor. Em memória de meu irmão, que aonde quer que esteja, está torcendo por mim. Amo vocês.

Aos amigos, que me acolheram nos momentos difíceis e que me incentivaram a não desistir.

Agradecer meu orientador, Prof. Dr. Samuel da Silva Feitosa, pelas revisões e perseverança. Por aceitar esse desafio comigo e me guiar na direção correta.

Aos membros da banca por aceitar fazer parte desse passo de minha vida, Prof. Me. Adriano Sanick Padilha e Prof. Dr. Felipe Grando.

“Não desanimeis jamais, embora venham ventos contrários.”

(Santa Madre Paulina)

RESUMO

O hemograma completo, exame microscópico de amostras de sangue humano, é um exame muito utilizado para avaliar o estado de saúde, alterações no organismo e diagnosticar doenças em pacientes. O exame consiste em analisar os elementos que compõem o sangue, contagem e classificação da parte vermelha (glóbulos vermelhos) e da parte branca (glóbulos brancos e plaquetas). Embora corriqueiro, o hemograma é realizado em laboratórios médicos por profissionais especializados, demandando muito tempo e alto custo em equipamentos e manutenção. Como alternativa, este projeto propõe um modelo computacional de *Deep Learning* capaz de classificar automaticamente as células brancas em imagens de amostras de sangue. Com ferramentas de Rede Neural Convolutiva, o modelo proposto alcançou a precisão de 97% na classificação de células brancas, utilizando base de dados pública e rotulada. Porém ao aplicar o modelo em imagens fora do padrão treinado, a precisão não se manteve. Para a apresentação dos dados de classificação, foi desenvolvido um protótipo web onde o profissional pode selecionar uma imagem para classificação e visualizar os dados reconhecidos para tomada de decisão na elaboração de um hemograma completo.

Palavras-chave: Redes Neurais Artificiais Convolucionais. Hemograma. Classificação. Células Brancas.

ABSTRACT

The complete blood count, a microscopic exam of human blood samples, is a test commonly used to assess state of health, organism changes and diagnose diseases in patients. The exam consists of analyzing the blood components, counting and classifying the elements between red part (red blood cells) and white part (white blood cells and platelets). The complete blood count is performed in medical laboratories by specialized professionals, demanding a lot of time and high cost equipment and maintenance. As an alternative, this project proposes a Deep Learning computational model, capable of classifying white cells in blood sample images automatically. Utilizing Convolutional Neural Network tools, the proposed model achieved 97% accuracy in white cell classification, using a labeled public testbase. However, when applying the model to images outside the trained pattern, the accuracy was not maintained. A web prototype was developed to the present the classified data and where the professional can select an image to assert the recognized data for decision making in the elaboration of a complete blood count.

Keywords: Convolutional Artificial Neural Network, Complete Blood Count, Classification, White Blood Cell.

LISTA DE ILUSTRAÇÕES

Figura 1 – Amostra de sangue sob microscópio.	21
Figura 2 – Valores Hematológicos de Referência - Parte Vermelha - Adultos.	22
Figura 3 – Valores Hematológicos de Referência - Parte Vermelha - Crianças.	22
Figura 4 – Valores Hematológicos de Referência - Parte Branca	23
Figura 5 – Exemplo de dada tipo de Célula Branca.	24
Figura 6 – <i>Perceptron</i>	26
Figura 7 – <i>Artificial Neural Network</i>	26
Figura 8 – Processo da Camada Convolutacional.	28
Figura 9 – Processo de Aprendizado e Classificação por CNN.	28
Figura 10 – Exemplo de Imagens Rotuladas no <i>Dataset</i>	36
Figura 11 – Imagem de Monócito antes de ser recortada para treinamento.	37
Figura 12 – Imagem de Monócito recortada para treinamento.	37
Figura 13 – Perda e Acurácia por 100 Épocas	42
Figura 14 – Perda e Acurácia por 20 Épocas	42
Figura 15 – Resultado Matriz de Confusão.	43
Figura 16 – Métrica Multiclasses.	44
Figura 17 – Protótipo para Detecção de Células.	45
Figura 18 – Protótipo para Classificação das Células Brancas.	46
Figura 19 – Coluna de Classificação das Células Brancas no Protótipo.	47
Figura 20 – Classificação Eosinófilo.	48
Figura 21 – Sem classificação.	49
Figura 22 – Classificação Dupla.	50
Figura 23 – Predição para cada Classe.	51

LISTA DE TABELAS

Tabela 1 – Palavras chaves e seus sinônimos.	31
Tabela 2 – Bases de dados pesquisadas e artigos encontrados.	32
Tabela 3 – Regra de exclusão e quantidade de artigos excluídos.	32
Tabela 4 – Identificador, Título e Autores dos artigos incluídos.	33
Tabela 5 – Quantidade de Imagens por Rótulos na Base de Dados.	36
Tabela 6 – Mapeamento Diretório x Classes.	39
Tabela 7 – Separação do <i>dataset</i> para cada funcionalidade.	39
Tabela 8 – Camadas Criadas para o Modelo de Classificação.	41

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Problema de Pesquisa	18
1.2	Hipótese de Pesquisa	18
1.3	Objetivos	18
1.3.1	Objetivo Geral	18
1.3.2	Objetivos Específicos	18
1.4	Justificativa	19
1.5	Código-Fonte	19
2	FUNDAMENTAÇÃO TEÓRICA	21
2.1	Hemograma Completo	21
2.1.1	Eritrograma	21
2.1.2	Leucograma	23
2.1.2.1	Células Brancas	23
2.1.2.2	Plaquetas	24
2.2	Inteligência Artificial	24
2.2.1	Aprendizado de Máquina	25
2.2.1.1	Redes Neurais Artificiais	25
2.2.2	Deep Learning	27
2.2.2.1	<i>Convolutional Neural Network (CNN)</i>	27
2.2.3	Ferramentas Disponíveis	29
3	MAPEAMENTO DA LITERATURA	31
3.1	Mapeamento Sistemático	31
3.1.1	Método de Exclusão de Artigos	32
3.1.2	CrITÉrios de Inclusão	32
4	METODOLOGIA	35
4.1	<i>Dataset</i> de imagens de células brancas	35
4.2	Ferramentas Utilizadas	37
4.3	Modelo Proposto	38
4.4	Análise dos resultados	41
4.5	Experimento	44
5	CONCLUSÕES	53
	Referências	55

1 INTRODUÇÃO

Nos últimos anos, impulsionado pelos novos avanços tecnológicos em hardware e software, *Deep Learning* (DL) virou uma tendência na área do Aprendizado de Máquina. Partindo de métodos baseados em Rede Neurais Artificiais, *Deep Learning* geralmente faz uso de múltiplas camadas ocultas (*Hidden Layers*) e novos paradigmas de treinamento para conseguir analisar com mais detalhes os dados, intercalando camadas para abstrair informações com mais precisão (1).

Deep Learning busca atingir resultados, aprendendo com um grande conjunto de dados. Esses dados devem ser devidamente coletados e pré-processados para que o processo de aprendizagem tenha erro mínimo. Modelos de DL passam por uma fase de treinamento com uma massa de dados e uma fase de testes com outra massa de dados distinta, para evitar *overfitting* (2).

Na área da saúde, DL tem sido utilizado com frequência no auxílio da classificação de dados sobre novas doenças e tomada de decisão por sua precisão nos resultados. A ferramenta também é muito utilizado para identificação de padrões em imagens, sendo possível encontrar inúmeros trabalhos que a usam para interpretação de imagens, porém comumente associado a necessidade de rotular dados para treinamento (3).

O hemograma, exame microscópico de amostras de sangue humano, é um exame muito utilizado para avaliar o estado de saúde e diagnosticar doenças em pacientes. Realizado em laboratórios médicos por profissionais especializados, esse exame corriqueiro demanda tempo e alto custo em equipamentos específicos. Utilizar de uma ferramenta para auxiliar nesse processo, aumentando a velocidade, a precisão da análise e até mesmo diminuindo o custo, é um grande avanço para todos. Poder utilizar dessa ferramenta ao lado do paciente no leito e ter respostas com rapidez, são detalhes que podem salvar vidas (4).

O processo manual de contagem e classificação das células consiste na utilização de lâminas de vidro para espalhar o sangue e microscópio para visualização dos elementos. Um processo automatizado faz uso de equipamentos específicos para este fim, os quais possuem custo de aquisição e manutenção elevados. Para redução desses custos, podemos utilizar de algoritmos de *Deep Learning* para automatização de extração de dados importantes à tomada de decisão, sem necessidade do uso do equipamento.

Nesse trabalho busca-se dar continuidade ao trabalho implementado por Anthony (2), adicionando a funcionalidade de classificação das células brancas presentes na amostra de sangue. A pesquisa (2) consiste em detectar as células vermelhas e brancas em uma imagem de amostra de sangue. Partindo de uma amostra de sangue, a ferramenta detecta as células com coordenadas, sendo possível a extração da célula para utilização posterior.

Partindo dessa detecção, a nova funcionalidade desenvolvida consiste em extrair e analisar a célula branca na imagem de amostra de sangue a fim de classificar seu tipo. A classificação dessas células será realizada por um modelo de DL, que será treinado utilizando um *dataset*

(grande conjunto de imagens) de células brancas de sangue humano em amostras de sangue. A partir desse treinamento, o modelo é capaz de classificar o tipo das células presentes na amostra.

1.1 PROBLEMA DE PESQUISA

Com o aumento na demanda por exames laboratoriais em tempos de pandemia e incerteza sanitária, é possível utilizar um modelo computacional que faça a classificação das células brancas de uma amostra de sangue, a fim de agilizar a produção de hemogramas?

1.2 HIPÓTESE DE PESQUISA

Com a utilização de um modelo computacional, treinado para a classificação de células brancas em imagens de amostras de sangue, podemos prover artifícios úteis para a elaboração de hemogramas com agilidade e precisão, deixando o processo mais rápido, barato e produtivo.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

O objetivo geral do trabalho é treinar um modelo computacional a ponto de ser capaz de classificar as células brancas em uma amostra de sangue com eficiência e agilidade, apresentando dados confiáveis para auxiliar o profissional da saúde na geração de um hemograma.

1.3.2 Objetivos Específicos

- Realizar um estudo e mapeamento sobre o tema para encontrar trabalhos relevantes na literatura sobre utilização de *Deep Learning* para classificação de células brancas em imagens de amostras de sangue.
- Buscar bases de dados disponíveis que contenham imagens de células brancas em amostras de sangue voltadas para o problema, para serem utilizadas como base de treinamento para o modelo de classificação das células.
- Pré-processar as imagens das bases de dados para montar um conjunto de dados homogêneos para o treinamento do modelo computacional de classificação.
- Desenvolver metodologia para extrair células brancas de imagens de amostras de sangue para serem classificadas.
- Desenvolver e treinar um modelo computacional de *Deep Learning* para classificação de células brancas em imagens de amostras de sangue.

- Desenvolver um protótipo funcional para classificação de células brancas através do modelo computacional treinado.
- Unir o protótipo de classificação de células brancas à aplicação de contagem de células já desenvolvida por Anthony em (2), para que possa ser usado para auxiliar profissionais da saúde.

1.4 JUSTIFICATIVA

O foco deste trabalho é apresentar uma forma de agilizar o processo de elaboração de um hemograma, utilizando um modelo computacional baseado em *Deep Learning* para a classificação das células brancas em uma imagem de amostra de sangue.

O presente trabalho contribui com a tipificação de células brancas para a ferramenta já produzida (2), apresentando dados úteis para a geração de hemogramas de forma rápida, confiável e assertiva.

Para os profissionais da área da saúde a ferramenta será útil à tomada de decisão rápida e eficiente.

Em tempos de crise sanitária a área da saúde é a mais afetada, tanto nos custos de matéria prima quanto na demanda de profissionais capacitados. Para contrapor esse efeito, é preciso alternativas aos processos monótonos e de extrema importância no dia a dia do profissional.

Frisa-se que os custos associados a produção de um hemograma são elevados devido à necessidade de compra e manutenção de maquinários especializados (5). Esse trabalho possibilitará a realização de análise laboratorial sem a necessidade de tais equipamentos.

A utilização de *Deep Learning* na área da saúde está em constante crescimento e tende a desenvolver tecnologias que substituam os profissionais nas tarefas mais simples, todavia, esse trabalho tem como finalidade a criação de uma aplicação, utilizando *Deep Learning* para o treinamento, que auxiliará o profissional da saúde na tomada de decisão e geração de hemograma, substituindo apenas o trabalho monótono e demorado.

1.5 CÓDIGO-FONTE

O código-fonte do protótipo apresentado neste trabalho foi desenvolvido na linguagem Python, utilizando a biblioteca e os recursos do Tensorflow e Keras em conjunto com os módulos de computação paralela da NVIDIA, CUDA e cuDNN. Para acesso aos detalhes do código do modelo implementado, pode ser acessado o repositório do Github (6).

2 FUNDAMENTAÇÃO TEÓRICA

2.1 HEMOGRAMA COMPLETO

Hemograma Completo (*Complete Blood Count*) é o exame mais comum realizado em laboratórios. É altamente usado para análise não invasiva de sintomas de anemia, cânceres, infecções, hemorragias, alergias e deficiências imunológicas, pelo fato de ser um exame altamente informativo sobre a saúde do paciente.

Nesse exame se realiza uma análise de amostras de sangue, obtidas através de microscópio, que envolve a contagem das diferentes células sanguíneas. Separado em duas principais etapas, o hemograma obtém dados relacionados as partes branca e vermelha do sangue, por processos chamados de Leucograma e Eritrograma especificamente (7, 8).

2.1.1 Eritrograma

Eritrograma pode revelar alguns tipos essenciais de alterações no sistema de produção do material vermelho do sangue, como aumento na produção de glóbulos vermelhos e anemias. O material vermelho corresponde as células vermelhas presentes no sangue, também chamadas de hemácias. De formato em discos circulares e sem núcleo, estão presentes em grande quantidade (8). É através das células vermelhas que é realizado o transporte de oxigênio e gás carbônico pelo sistema sanguíneo, permitindo as trocas gasosas necessárias ao metabolismo (9). Na Figura 1 estão rotuladas por RBC (*Red Blood Cell*).

Figura 1 – Amostra de sangue sob microscópio.



Fonte: Dataset - Shenggan: https://github.com/Shenggan/BCCD_Dataset/blob/master/example.jpg

O Eritrograma tem como objetivo principal a análise sobre a parte vermelha do sangue. E, dessa análise, são obtidas informações importantes sobre o paciente.

Os valores obtidos para cada item abaixo são comparados com as tabelas de referência nas Figuras 2 e 3, que variam conforme idade e sexo do indivíduo e localização geográfica (altitude) (9).

As informações mais comuns de serem obtidas são: A contagem geral das células vermelhas com escala de milhões/mm³; quantidade de hemoglobina em escala de g/dL; o Volume Corpuscular Médio (VCM) que indica a média do tamanho das hemácias por fL (fentolitro); a Hemoglobina Corpuscular Média (HCM), quantidade média de hemoglobina presente nas hemácias por pg (picoGramas); a Concentração de Hemoglobina Corpuscular Média (CHCM) indica a porcentagem de hemoglobina em uma amostra de hemácias; e a Amplitude de Distribuição dos Glóbulos Vermelhos (RDW - Red Cell Distribution Width) que aponta a variação do tamanho das hemácias.

Figura 2 – Valores Hematológicos de Referência - Parte Vermelha - Adultos.

Valores hematológicos de referência em adultos			
		Homens	Mulheres
Hemácias	x 10 ¹² /L	5,00 ± 0,5	4,3 ± 0,5
Hemoglobina	g/dL	15,0 ± 2,0	13,5 ± 1,5
Hematócrito	(%)	45 ± 5	41 ± 5
Leucócitos	x 10 ⁹ /L	7,0 ± 3,0	
VGM	fL	92 ± 9	
HGM	pg	29,5 ± 2,5	
CHGM	g/dL	33 ± 1,5	
RDW	CV (%)	12,8 ± 1,2	
	SD (fL)	42,5 ± 3,5	
Plaquetas	x 10 ⁹ /L	150 - 400	

Fonte: PNCQ (10)

Figura 3 – Valores Hematológicos de Referência - Parte Vermelha - Crianças.

Valores hematológicos de referência – série vermelha						
Idade	Hemoglobina (g/dL)	Hematócrito (%)	Hemácias (x 10 ⁶ /μL)	VGM (fL)	HGM (pg)	CHGM (g/dL)
Nascimento	18,0 ± 4,0	60 ± 15	6,0 ± 1,0	110 ± 10	34 ± 3	33 ± 3
3 dias	18,0 ± 3,0	56 ± 11	5,3 ± 1,3	105 ± 13	34 ± 3	33 ± 4
1 mês	14,0 ± 2,5	43 ± 10	4,2 ± 1,2	104 ± 12	33 ± 3	33 ± 4
2 meses	11,2 ± 1,8	35 ± 7	3,7 ± 0,6	95 ± 8	30 ± 3	32 ± 5
3 - 6 meses	12,6 ± 1,5	35 ± 5	4,7 ± 0,6	76 ± 8	27 ± 3	33 ± 3
1 ano	12,6 ± 1,5	34 ± 4	4,5 ± 0,6	78 ± 6	27 ± 2	34 ± 2
2 - 6 anos	12,6 ± 1,5	37 ± 3	4,5 ± 0,6	81 ± 6	27 ± 3	34 ± 3
6 - 12 anos	12,5 ± 1,5	40 ± 5	4,5 ± 0,6	86 ± 9	29 ± 4	34 ± 3

Fonte: PNCQ (10)

2.1.2 Leucograma

O Leucograma tem como finalidade avaliar o sistema de defesa do organismo e com isso verificar se há alguma inflamação ou infecção. Essa verificação ocorre analisando a parte branca do sangue, pontuando atributos-chave como a classificação e contagem de diferentes células brancas.

Assim como no Eritrograma, no Leucograma são levantados alguns itens principais para análise e comparação com valores de referência, reproduzidos na Figura 4

Os principais itens para análise de um Leucograma são: Contagem de Leucócitos (células brancas); Classificação e contagem de cada tipo de leucócito: Linfócitos, Monócitos, Neutrófilos, Eosinófilos, Basófilos; Quantidade de Plaquetas.

Figura 4 – Valores Hematológicos de Referência - Parte Branca

Valores hematológicos de referência – série branca							
Idade	Leucócitos/uL	Neutrófilos/uL	Linfócitos/uL	Monócitos/uL	Eosinófilos/uL	Basófilos/uL	Plaquetas/uL
Nascimento	18.000 10.000 – 26.000	4.000 – 14.000	3.000 – 8.000	500 – 2.000	100 – 1.000	20 – 100	150 – 450
3 dias	15.000 7.000 – 22.000	3.000 – 5.000	2.000 – 8.000	500 – 1.000	100 – 2.000	20 – 100	210 – 500
1 mês	12.000 5.000 – 19.000	3.000 – 9.000	3.000 – 16.000	300 – 1.000	200 – 1.000	20 – 100	210 – 650
2 meses	10.000 5.000 – 15.000	1.000 – 5.000	4.000 – 10.000	400 – 1.200	100 – 1.000	20 – 100	210 – 650
3 - 6 meses	12.000 6.000 – 18.000	1.000 – 6.000	4.000 – 12.000	200 – 1.200	100 – 1.000	20 – 100	200 – 550
1 ano	11.000 6.000 – 16.000	1.000 – 7.000	3.500 – 11.000	200 – 1.000	100 – 1.000	20 – 100	200 – 550
2 - 6 anos	10.000 5.000 – 15.000	1.500 – 8.000	6.000 – 9.000	200 – 1.000	100 – 1.000	20 – 100	200 – 450
6 - 12 anos	9.000 5.000 – 13.000	2.000 – 8.000	1.000 – 5.000	200 – 1.000	100 – 1.000	20 – 100	180 – 400
Adultos	7.000 4.000 - 10.000	2.000 – 7.000	1.000 – 3.000	200 – 1.000	20 – 500	20 – 100	150 – 400

Fonte: PNCQ (10)

2.1.2.1 Células Brancas

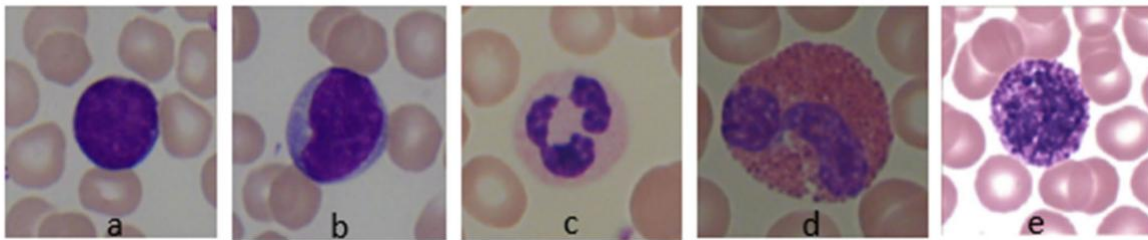
Em grande quantidade na parte branca do sangue, os glóbulos brancos, *White Blood Cell* (WBC), são as células responsáveis pela defesa do organismo contra ameaças do corpo humano presentes no sistema sanguíneo. Através da fagocitose, que é um processo de englobamento de partículas sólidas pelas células, são realizadas ações de defesa contra a invasão de fragmentos estranhos (9).

As células brancas são separadas em cinco principais tipos com características visuais únicas e funções estritamente definidas.

- A) **Linfócitos:** Responsáveis pela imunidade do corpo humano, combate a vírus e tumores e produção de anticorpos. Quando em grande quantidade, podem indicar uma infecção viral, HIV, leucemia ou rejeição de um órgão transplantado.

- B) **Monócitos:** Células Brancas que sinalizam a presença de organismos estranhos e aciona demais células, fagocitando microrganismos invasores. Atuam contra vírus e bactérias sem distinção. São capazes de entrar no tecido conjuntivo frouxo, onde conseguem se desenvolver em grandes células do pulmão e fígado.
- C) **Neutrófilos:** Células Brancas mais abundantes. Capazes de deixar os vasos sanguíneos e entrar nos tecidos, onde conseguem realizar a defesa do organismo, fagocitando partículas estranhas. Responsável pelo combate a infecções, podendo ser indicativo de infecção por bactérias quando os valores encontram-se elevados. Os bastões são os neutrófilos jovens e que normalmente são encontrados no sangue quando há infecções em fase aguda.
- D) **Eosinófilos:** Células Brancas responsáveis na defesa contra parasitas, também presentes em grande quantidade durante reações alérgicas.
- E) **Basófilos:** Células Brancas que atuam em reações alérgicas e na coagulação do sangue contra inflamações.

Figura 5 – Exemplo de dada tipo de Célula Branca.



Fonte: Khan et al. (11).

2.1.2.2 Plaquetas

Também presentes na parte branca do sangue estão as Plaquetas. Mesmo sendo os menores componentes do sangue, possuem grande responsabilidade na resposta fisiológica para a prevenção e interrupção de sangramentos e hemorragias (8). Percebe-se na Figura 1 um elemento com coloração azul e rotulada de *Platelets*.

A utilização de computadores para diagnóstico do hemograma são essenciais, logo, a automatização desse processo pode ser computadorizada. Baseando-se em tecnologias emergentes de visão computacional e inteligência artificial, grande parte do processo de elaboração de hemograma pode ser automatizado.

2.2 INTELIGÊNCIA ARTIFICIAL

No presente trabalho o foco se dará na abordagem de *Deep Learning* como modelo computacional para classificação das células brancas. Esse modelo tem grande popularidade

para esse objetivo por usar múltiplas camadas de processamento ocultas (*Hidden Layers*) capazes de generalizar e obter mais detalhes sobre as imagens analisadas. *Deep Learning* é uma subárea de *Machine Learning*, que por sua vez, é uma subárea da Inteligência Artificial (IA). IA é uma abordagem computacional para resolução de problemas de forma automatizada para diversos problemas e tarefas do cotidiano, sem a necessidade de um usuário ou especialista. Além de atender demandas em automação de atividades da indústria, também está presente em finalidades básicas pessoais do dia a dia. O intuito de métodos de IA é ensinar máquinas e computadores a serem capazes de analisar e interpretar dados. Utilizando um grande volume de dados e diversas técnicas, IA permite que os computadores sejam ensinados a reconhecer padrões em dados, semelhante aos seres humanos (1).

2.2.1 Aprendizado de Máquina

Com a necessidade de analisar grande quantidade de dados o Aprendizado de Máquina, em inglês *Machine Learning* (ML), surge como solução promissora. Com mínimo de interação humana, um modelo de ML monta um sistema analítico para as tomadas de decisão, capazes de aprender analisando os dados e generalizar para identificação e análise de dados genéricos (1).

ML é separado em duas principais metodologias que atendem determinados problemas: Aprendizado Supervisionado e Não Supervisionado. No aprendizado supervisionado conhecemos o resultado esperado, porém não conhecemos como chegar até ele. Diferente do aprendizado não supervisionado, onde não há interação e *feedback* de confirmação sobre o resultado. Uma outra metodologia seria a combinação dos dois métodos. Para treinamento do modelo de ML usa-se aprendizado não supervisionado e após isso usa-se aprendizado supervisionado para efetuar pequenas correções no modelo. Aprendizado supervisionado sobressai em abordagens de classificação, onde se tem conhecimento prévio sobre os rótulos de cada classe sendo analisada (12, 13).

2.2.1.1 Redes Neurais Artificiais

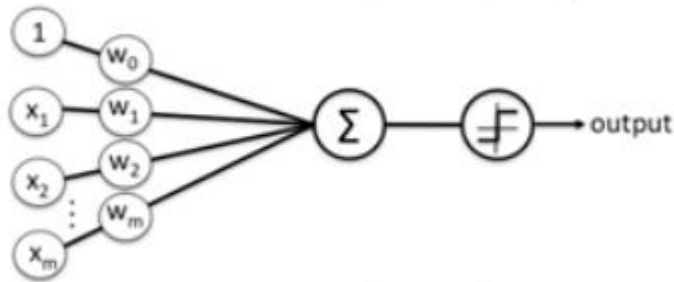
Outra importante estrutura de ML, a Rede Neural Artificial, em inglês *Artificial Neural Network* (ANN), utiliza uma abordagem inspirada no cérebro humano. O cérebro humano é composto por milhões de neurônios interligados, formando uma rede neural biológica. Cada informação recebida altera essa estrutura de acordo com a informação e é acionado conforme o estímulo químico daquela informação.

Uma rede neural artificial também contém neurônios, chamados de *perceptrons*, que são interligados. Cada *perceptron* é capaz de receber dados de entrada, *inputs*, e produzir uma saída, *output*. Conforme o cérebro vai aprendendo, as conexões entre os neurônios são criadas, alteradas ou removidas. Similar a isso, o neurônio artificial muda os pesos da informação dos *inputs*, com intuito de minimizar o erro na saída. Esses pesos de ativação podem ser vistos

como critério de prioridade entre as informações recebidas (12).

A Figura 6 apresenta a estrutura de um perceptron, onde o x representa os *inputs*, o w representa os pesos de cada informação, o símbolo Σ é o somatório dos pesos da entrada que é passado para a função de ativação F , o resultado da função é retornado em *output*.

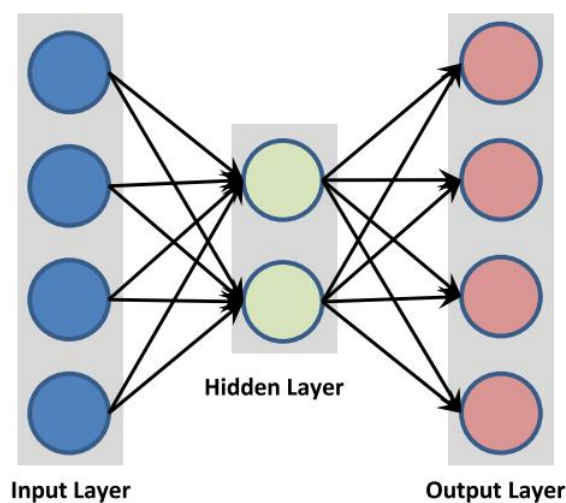
Figura 6 – *Perceptron*.



Fonte: Mahesh (12)

A ligação de múltiplos *perceptrons* forma uma Rede Neural Artificial. Definida por múltiplas camadas de *perceptrons*, geralmente uma camada de *inputs*, uma camada de *output* e camadas intermediárias chamadas de *hidden layers* (HL). Cada HL pode conter um número variável de *perceptrons* e é responsável por processar a informação recebida pela função de ativação e transmitir para a saída, conforme a Figura 7.

Figura 7 – *Artificial Neural Network*.



Fonte: Ravì et al. (3)

É importante ressaltar o treinamento de uma ANN, onde é feita uma predição dos dados de entrada usando valores aleatórios para as ligações entre os *perceptrons*. Os resultados são comparados com os valores esperados para saber se está obtendo sucesso ou não. A diferença entre o resultado obtido e o resultado esperado é chamado de erro. O erro é usado por um algoritmo de correção e aprendizagem que irá modificar os pesos para priorizar as informações e minimizar o erro. Esses algoritmos são conhecidos como *backpropagation*, propagação reversa, porque a correção começa das camadas finais e vai propagando até as camadas iniciais. O erro de cada camada influencia na alteração dos pesos da camada anterior. Por fim é repetido esse processo inúmeras vezes até que o erro entre o resultado final e o esperado esteja aceitável MLAlgMahesh, DPAlgArch.

2.2.2 *Deep Learning*

Alguns problemas atuais não são possíveis de serem tratados utilizando os modelos tradicionais de ML, pois o aprendizado acontece através de reconhecimento de padrões com base nos dados que não podem ser utilizados antes de serem preparados e adaptados para cada modelo. Em geral, a máquina não é capaz de aprender sem interferência humana, pois precisa dos dados processados para o padrão do modelo. O *Deep Learning* (DL) surge como conceito aprimorado para uma NN, criando assim o modelo *Deep Neural Network* (DNN). DNN é uma rede neural profunda, que contém modelos mais complexos, com maior número de neurônios, camadas ocultas de alta complexidade e interconexões específicas (12, 13).

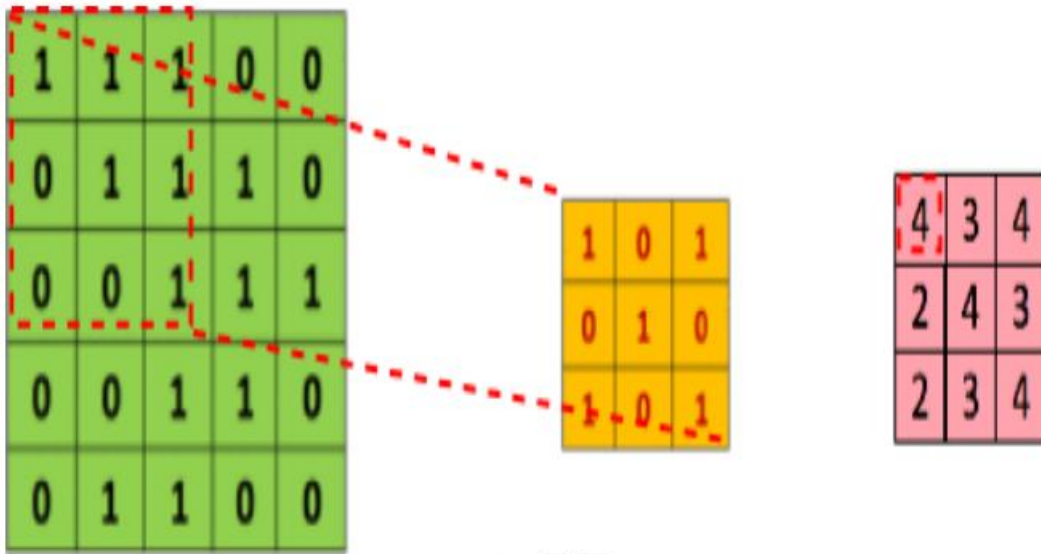
2.2.2.1 *Convolutional Neural Network* (CNN)

O alto número de camadas e conexões em uma DNN possibilitou o surgimento de novas propostas para atender problemas específicos. Como reconhecimento e interpretação de imagens, que é o objeto de estudo desse trabalho. Para esse problema será abordada a solução conhecida por CNN.

As Redes Neurais Convolucionais, CNNs, são arquiteturas desenvolvidas especificamente para o reconhecimento de imagens, onde cada imagem é traduzida para uma matriz de três dimensões contendo altura, comprimento e cores ($X_{px} \times Y_{px} \times 3_{cores}$ (RGB)), reduzindo drasticamente a quantidade de parâmetros e o tempo de treinamento da rede, sem a necessidade de interligar totalmente todas as camadas ocultas. Essa matriz então é usada como dados de entrada para as camadas iniciais de filtragem e identificação de características. Essa camada de filtragem é chamada de convolucional, que faz a multiplicação das matrizes da imagem de entrada com uma outra matriz de filtragem pré-determinada (14).

Na Figura 8 podemos identificar a matriz de cor verde que corresponde a matriz da imagem (uma das dimensões), uma matriz menor amarela que atua como matriz de filtro e por último uma matriz vermelha que é o resultado da filtragem usando uma função de somatório. A

Figura 8 – Processo da Camada Convolutiva.

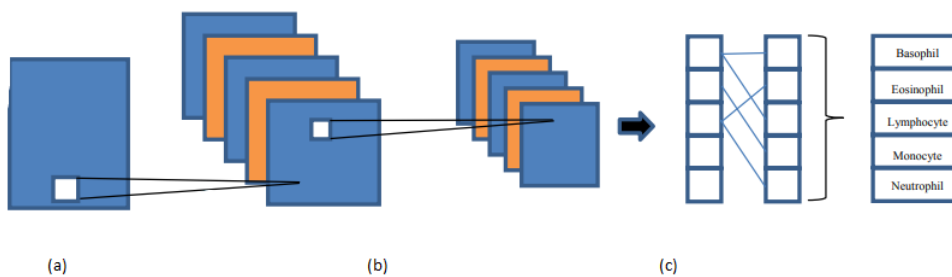


Fonte: Tiwari et al. (14)

matriz vermelha é calculada utilizando o produto de matrizes, movendo a área pontilhada por toda matriz da imagem.

Uma vez que a CNN identificou uma característica, independente da posição na imagem onde foi identificada, ela será capaz de generalizar e identificar a mesma característica em qualquer posição da imagem de entrada (15).

Figura 9 – Processo de Aprendizado e Classificação por CNN.



Fonte: Macawile et al. (15)

A Figura 9 resume o processo de aprendizagem e classificação de uma CNN, onde a sessão *a* é a imagem de entrada crua sem filtro. A sessão *b* é a camada de filtragem convolutiva de obtenção das características, que pode ser repetida até que o resultado seja de acordo com os parâmetros de entrada para camada oculta na sessão *c*. E, por último, temos a classificação conforme as características extraídas.

2.2.3 Ferramentas Disponíveis

Esses modelos de *Deep Learning* já estão disponíveis em ferramentas públicas como *TensorFlow* e *KERAS*, porém as redes disponíveis são treinadas com um intuito de generalização. São modelos treinados com dados sem área específica (16, 17, 2).

Criada pela Google, o *TensorFlow* é caracterizado por ser uma interface de código aberto focado em ML. Com foco em redes neurais artificiais, seus módulos permitem que problemas de várias áreas possam ser reproduzidas. Existem módulos de treinamento, validação e análise de resultados (16).

Já incluso na ferramenta do *TensorFlow*, o *KERAS* permite uma experimentação e prototipação rápida, com foco em *Deep Learning* de fácil utilização, modular e extensível. Também possui módulos voltados à CNN (17).

Como o tratamento dos dados de entrada e o processamento da CNN são tarefas com alto preço computacional, são fatores determinantes para o tempo de treinamento do modelo. Com o auxílio dessas ferramentas, as tarefas de tratamento e treinamento se tornam mais fáceis e ágeis, pois possibilitam a utilização da placa gráfica (GPU) para treinamento, ao invés de utilizar o processador (CPU).

3 MAPEAMENTO DA LITERATURA

Nessa seção serão revisados alguns trabalhos publicados que utilizam *Deep Learning* classificação de células brancas. Embora muitos desses trabalhos sejam publicados em plataformas de acesso restrito (11), artigos relevantes foram encontrados para análise.

O mapeamento sistemático da literatura serviu para selecionar os trabalhos mais relevantes e mais recentes sobre os temas que serão abordados nesse trabalho. O processo de pesquisa e seleção dos trabalhos relacionados foi realizado com a intenção de encontrar trabalhos com propostas de agilizar a extração e classificação de células brancas em amostras de sangue. Além dos principais trabalhos obtidos, outro objetivo foi verificar métodos de *Deep Learning* mais utilizados e com melhor desempenho na extração e na classificação de células brancas em amostras de sangue, para obtenção de informações dos tipos de células brancas presentes em uma amostra de sangue em hemogramas.

3.1 MAPEAMENTO SISTEMÁTICO

No processo de pesquisa dos artigos na literatura, foi realizado a montagem de uma *string* de busca nas principais bibliotecas e repositórios de artigos. Para a busca e seleção dos artigos foi utilizada a ferramenta Parsifal¹, que auxilia na definição da *string* de busca, gerencia os artigos encontrados e suas fontes, e permite a manipulação dos artigos por métodos de aprovação e reprovação.

Partindo do problema de pesquisa descrito no presente trabalho, foram extraídas palavras chaves e sinônimos para o direcionamento da pesquisa. Na Tabela 1 podem ser vistas as palavras chaves extraídas e seus sinônimos utilizados.

Tabela 1 – Palavras chaves e seus sinônimos.

Palavra Chave	Sinônimos
<i>Blood Analysis</i>	<i>Blood Sample, Complete Blood Count, White Blood Cell</i>
<i>Classification</i>	<i>Interpretation, Recognition, Segmentation</i>
<i>Deep Learning</i>	<i>Artificial Intelligence, Computer Vision, Machine Learning</i>

Fonte: Elaborado pelo autor.

A partir das palavras chaves e dos sinônimos explícitos a ferramenta *Parsifal* dispõe de uma funcionalidade de geração de *string* de busca. Com o auxílio dessa funcionalidade foi gerada então a seguinte *string* de busca para pesquisa em bases de dados digitais de artigos:

("Blood Analysis"OR "Blood Sample"OR "Complete Blood Count"OR "White Blood Cell") AND ("Classification"OR "Interpretation"OR "Recognition"OR "Segmentation") AND ("Deep Learning"OR "Artificial Intelligence"OR "Computer Vision"OR "Machine Learning")

¹ <https://parsif.al/>

Na Tabela 2 são listadas as bases de dados em que os artigos foram selecionados, quantidade de artigos encontrados em cada base para uso no trabalho. Além da *string* de busca também foram filtrados apenas artigos mais relevantes entre os anos de 2020 e 2022.

Tabela 2 – Bases de dados pesquisadas e artigos encontrados.

Base de Dados	Artigos Encontrados
<i>IEEE Digital Library</i>	32
<i>ACM Digital Library</i>	45

Fonte: Elaborado pelo autor.

3.1.1 Método de Exclusão de Artigos

Após a seleção dos artigos encontrados com o auxílio da *string* de busca, os 77 artigos mais relevantes encontrados passaram por uma etapa de exclusão, seguindo algumas regras estabelecidas, descritas na Tabela 3.

Tabela 3 – Regra de exclusão e quantidade de artigos excluídos.

Regra de Exclusão	Quantidade Excluídos
Estudo não faz parte da área pesquisada	58
Resultados fora da área da computação	11
Duplicatas	0

Fonte: Elaborado pelo autor.

Partindo de um total de 77 artigos relacionados as área de pesquisa, na etapa de exclusão foram rejeitados um total de 69 artigos, resultando em 8 artigos aprovados para utilização no trabalho. Podemos observar que uma grande parte foi classificada como “Estudo não faz parte da área pesquisada”, pois tinham relação com uso de algoritmos de *machine learning*, *deep learning* e *artificial intelligence*, porém não tinham aplicação direta ou descrição sobre área pesquisada. Outros 11 artigos foram rejeitados pelo critério “Resultado fora da área da computação”. Significando que são artigos com relação a alguma área pesquisada porém sem resultados relacionados a computação. Para regra de “Duplicatas” não houve rejeição.

3.1.2 Critérios de Inclusão

Para aprovação dos artigos também foi criada uma etapa de Inclusão. Os artigos remanescentes após a exclusão forma classificados conforme regras abaixo:

1. Contexto sobre um dos temas;
2. Descreve sistema para elaboração de hemogramas;
3. Descreve sistema para classificação de células brancas;

4. Métodos de Deep Learning aplicados à Segmentação ou Classificação de Células Brancas.

Os critérios para inclusão de artigos formaram um guia para escolha de artigos que estão relacionados à recursos para auxiliar no desenvolvimento de um modelo computacional baseado em *Deep Learning* para segmentação e classificação de células brancas, a fim de agilizar a tomada de decisão na elaboração de um hemograma. Na Tabela 4 estão listados os artigos selecionados utilizando os critérios de inclusão, juntamente com um identificador, seu título e autores.

Tabela 4 – Identificador, Título e Autores dos artigos incluídos.

ID	Título	Autore(s)	Ano
A01	A Review on Traditional Machine Learning and Deep Learning Models for WBCs Classification in Blood Smear Images	Khan, Siraj and Sajjad, Muhammad and Hussain, Tanveer and Ullah, Amin and Imran, Ali Shariq	2021
A02	<i>An Automated Approach to White Blood Cell Classification Using a Lightweight Convolutional Neural Network</i>	Ridoy, Md. Alif Rahman and Islam, Md. Rabiul	2020
A03	<i>An improved deep learning based classification of human white blood cell images</i>	Siddique, Md. Abu Ismail and Aziz, Abu Zahid Bin and Matin, Abdul	2020
A04	<i>Analysis of WBC, RBC, Platelets Using Deep Learning</i>	Shinde, Srushti and Oak, Jui and Shrawagi, Kajal and Mukherji, Prachi	2021
A05	<i>Classification of blood cells into white blood cells and red blood cells from blood smear images using machine learning techniques</i>	Navya, K.T and Prasad, Keerthana and Singh, Brij Mohan Kumar	2021
A06	<i>Comparative analysis of White Blood Cells Classification using Deep Learning Architectures</i>	Kousalya, K. and Krishnakumar, B. and Mohana, R. S. and Karthikeyan, N.	2021
A07	<i>Leukocytes Segmentation and Classification in Digital Microscopic Images</i>	Hussain, Muhammad Abbas and Ahmad, Ibtihaj and Shaukat, Arslan and Islam, Zain UI	2021
A08	<i>Towards Efficient Segmentation and Classification of White Blood Cell Cancer Using Deep Learning</i>	Iqbal, Ashik and Ahmed, Md Faysal and Suvon, Md Naimul Islam and Shuvho, Sourav Das and Fahmin, Ahmed	2021

Fonte: Elaborado pelo autor.

O último passo realizado para o mapeamento sistemático da literatura foi uma análise dos artigos selecionados a fim de extrair os métodos mais utilizados daqueles com maior precisão, e até mesmo novos métodos não convencionais para segmentação e classificação de células brancas em imagens de amostra de sangue.

Os algoritmos mais utilizados são técnicas de *Deep Learning*, mais precisamente Rede Neural Convolutacional (*Convolutional Neural Network* - CNN), que é o método mais utilizado entre os artigos selecionados. Presente nos artigos A01 (11), A02 (18), A03 (19), A04 (20), A06 (21) e A08 (22), o método de CNN é comumente usado para análise de imagens pela sua alta fidelidade e precisão, ótima para segmentação e classificação de células brancas em imagens de amostra de sangue. *Support Vector Machine* (SVM) está também entre os métodos mais utilizados. Alcançando, em alguns casos, maior precisão na classificação em imagens comparado a CNN. Os artigos A05 (23) e A07 (24) descrevem as capacidades e assertividade do método SVM em segmentação e classificação em imagens.

4 METODOLOGIA

No presente trabalho foi desenvolvido um modelo utilizando Rede Neural Artificial com o objetivo de classificação dos tipos de células brancas em amostras de sangue. Esse modelo fez uso de um *dataset* de imagens de células brancas públicas e ferramentas comuns na prototipação de modelos computacionais.

4.1 DATASET DE IMAGENS DE CÉLULAS BRANCAS

Para a criação do modelo de *Deep Learning*, uma base de dados, ou um *dataset*, é de extrema importância, já que é a partir dele que o modelo será capaz de identificar características para generalizar e classificar os tipos de células brancas. O *dataset* escolhido precisou atender alguns requisitos para que o treinamento do modelo fosse efetivo.

Por se tratar de um aprendizado supervisionado, o *dataset* utilizado necessitou que cada imagem tenha um rótulo atrelado, correspondente ao tipo da célula a ser classificada. Para melhorar a efetividade do modelo, uma grande quantidade de imagens foi indispensável para os processos de treinamento, validação e teste. A qualidade das imagens teve suma importância para processo de classificação. Uma imagem com muito ruído ou de baixa definição não ajudará o modelo a generalizar uma característica.

Com base nesses requisitos foi buscado em acervos públicos um *dataset* que atendesse as necessidades. O *dataset* publicado em (25) se enquadrou em todos os pontos e por isso foi selecionado.

Com um total de 17.092 imagens relacionadas a células brancas, todas coletadas de pacientes sem patologias sanguíneas, por profissionais do Hospital das Clínicas de Barcelona utilizando equipamento especializado. O *dataset* já é rotulado em oito grupos: Neutrófilos, Eosinófilos, Basófilos, Linfócitos, Monócitos, Eritroblastos, Plaquetas e Granulócitos Imaturos. O tamanho de cada imagem é de 360px x 363px, colorida e com formato *JPG*. Sendo o primeiro acervo público com grande quantidade de imagens de alta qualidade referentes a células sanguíneas (25).

Esse *dataset* foi ideal para a área de pesquisa. Com esse grande conjunto de dados rotulados e de alta qualidade, foi possível treinar e testar um modelo robusto para classificar diferentes tipos de células brancas.

Para o objetivo de classificar os cinco principais tipos de células brancas, são eles Neutrófilos, Eosinófilos, Basófilos, Linfócitos e Monócitos, as imagens e os grupos Plaquetas, Granulócitos Imaturos e Eritroblastos serão removidos da base de dados. Desse modo temos apenas os rótulos necessários para o objetivo.

A rotulagem do *dataset* foi feita através de diferentes diretórios. Para cada rótulo existe um diretório específico, e dentro dele suas respectivas imagens. A Tabela 5 descreve a estrutura da base de dados após remoção dos grupos, totalizando 10298 imagens nos cinco grupos

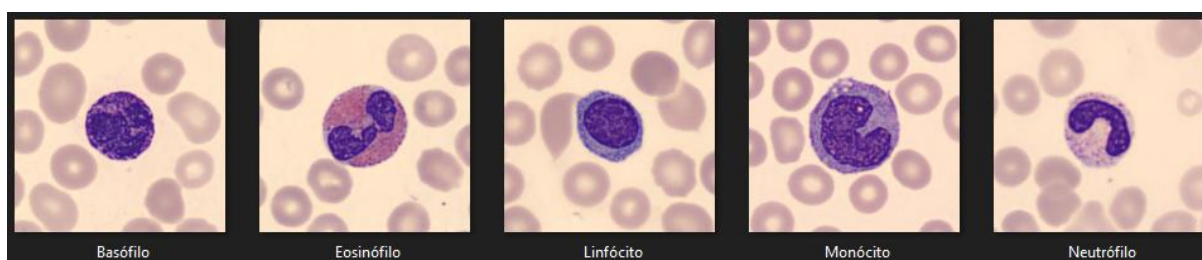
desejados. E a Figura 10 mostra um exemplo de cada rótulo.

Tabela 5 – Quantidade de Imagens por Rótulos na Base de Dados.

Rótulo / Diretório	Quantidade de Imagens
Basófilo	1218
Eosinófilo	3117
Linfócito	1214
Monócito	1420
Neutrófilo	3329

Fonte: Elaborado pelo autor.

Figura 10 – Exemplo de Imagens Rotuladas no *Dataset*



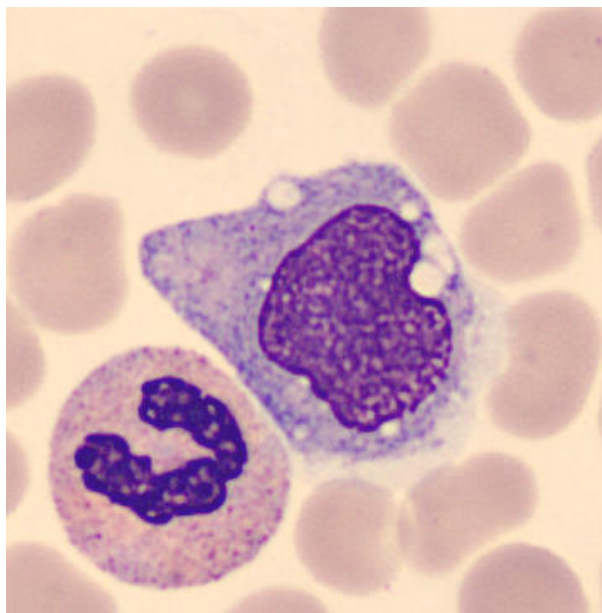
Fonte: Reproduzido de: Acevedo et al. (25)

Como vemos nos exemplos, a célula rotulada aparece com muita informação irrelevante para o nosso modelo. A maior parte da imagem é composta por parte vermelha do sangue, o que não é ideal para o nosso treinamento, já que poderá afetar na aprendizagem. Para classificar uma célula branca precisamos extraí-la do meio dessas informações. Por isso iremos tratar esse *dataset* para se comportar adequadamente ao padrão utilizado na detecção por CRUZ (2).

Se a detecção da célula branca não alcançar uma taxa de confiança de 50% no processo de detecção (2), ela não será classificada. Para as células que passarem da taxa de corte, foi recortada da imagem original respeitando a caixa de detecção do modelo. A imagem recortada será enviada ao modelo de classificação para que seja classificado em uma das classes treinadas.

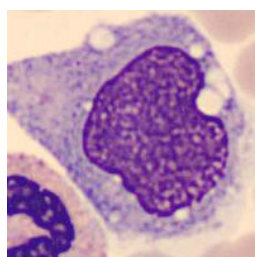
Para o tratamento do *dataset*, foi conferido que na grande maioria das imagens a célula rotulada está no centro. Podendo assim ser recortada e ajustada para um tamanho adequado ao treinamento. Conforme visualizado na Figura 11, podemos perceber que existem duas células de classes diferentes, porém a que está no centro é a rotulada corretamente como Monócito. E na Figura 12 podemos ver a imagem recortada para treinamento com tamanho 200x200 px.

Figura 11 – Imagem de Monócito antes de ser recortada para treinamento.



Fonte: Reproduzido de: (25)

Figura 12 – Imagem de Monócito recortada para treinamento.



Fonte: Elaborado pelo autor.

4.2 FERRAMENTAS UTILIZADAS

A biblioteca *Tensorflow* foi utilizada para o desenvolvimento do modelo de classificação, pois possui um grande acervo de utilitários que permite atender o problema levantado neste trabalho utilizando Inteligência Artificial. Os motivos para a escolha dessa biblioteca são, a documentação completa com exemplos de suas funcionalidades e atributos, e a alta utilização para problemas de Redes Neurais Artificiais (RNA) na literatura.

Nas versões atuais do *Tensorflow*, existe integração nativa com a ferramenta *Keras* como um módulo para prototipação de RNAs. Esse módulo contém inúmeras implementações que simplificam a construção de RNA, como camadas, funções de ativação, otimizadores e uma série de ferramentas que facilitam o manuseio de dados de imagem e texto para facilitar o desenvolvimento e escalabilidade, possibilitando a execução de seus modelos em *clusters* de placas de vídeo (GPU), destacando-se tanto na indústria quanto na comunidade de pesquisa (17).

As GPU's são fortemente recomendadas para a prototipação de um modelo de processamento de imagens, ainda mais com um grande volume, pois possuem um desempenho mais eficaz se comparado ao processador (CPU). Essas ferramentas tem integração homologada com GPU da marca *NVIDIA*, que para sua utilização no processamento é necessária a instalação das ferramentas *CUDA* e *cuDNN*, da mesma empresa.

CUDA é uma plataforma para realização de computação paralela, para fornecer aos desenvolvedores o melhor aproveitamento do processamento das GPU's e acelerar os processos de computação. Em combinação com *CUDA*, a *cuDNN* é uma biblioteca com foco em Redes Neurais Artificiais Profundas que permite utilizar estruturas de Redes Neurais Artificiais no processamento de algoritmos. O uso do *CUDA* associado ao *cuDNN*, aumenta a eficiência e também a qualidade do resultado (26, 27).

Com as ferramentas selecionadas e instaladas foi possível comprovar suas facilidades e efetividades. Com o processamento de funções de exemplo das ferramentas via GPU, foi possível confirmar que o treinamento de um modelo computacional para classificação de imagens tem um ganho considerável quando comparado ao processamento em CPU.

Com o dataset escolhido e as ferramentas instaladas foi iniciado o desenvolvimento do modelo de classificação de células brancas.

4.3 MODELO PROPOSTO

Nos trabalhos relacionados foi destacado a utilização de modelos pré treinados com imagens genéricas como modelo computacional para classificação, utilizando a técnica de *transfer learning*. Porém, após testes iniciais com um modelo treinado apenas com o *dataset* obtido, foi verificado que o resultado ficou acima do esperado, alcançando uma precisão de 94%. Por esse motivo, a técnica de *transfer learning* não foi utilizada, e sim, montado um modelo com treinamento com apenas imagens do *dataset* escolhido. Ficará como trabalhos futuros uma implementação e aperfeiçoamento utilizando *transfer learning*.

No modelo proposto foram utilizadas todas as imagens do *dataset* referentes aos grupos de células. Cada grupo tinha uma distribuição desparelha de imagens. A leitura do *dataset* foi feita pela ferramenta *Keras*, utilizando da função *image_dataset_from_directory*, que lê as imagens de um diretório, em ordem alfabética, e já faz o mapeamento do rótulo para classe pelo nome do diretório que está inserida. A Tabela 6 descreve o mapeamento para o formato atual do *dataset* que foi carregado.

Com o *dataset* carregado e rotulado corretamente, separamos os dados em dados de treinamento, dados de validação e dados de teste. Para a separação foi seguido o formato mais utilizado na literatura de 70-20-10, que consiste em 70% das imagens do *dataset* deve ser usado para função de treinamento, 20% para função de validação e correção durante o treinamento e os 10% restantes para função de testes.

Com a utilização da ferramenta *Keras*, o *dataset* é carregado em formato de lote, em

Tabela 6 – Mapeamento Diretório x Classes.

Rotulo / Diretório	Classe Mapeada
Basófilo	0
Eosinófilo	1
Linfócito	2
Monócito	3
Neutrófilo	4

Fonte: Elaborado pelo autor.

inglês *batch*. Cada *batch* é composto por uma quantidade de imagens aleatórias de cada classe. Um *batch* consiste em um array de duas posições de tamanho configurável, que por padrão é de 32 imagens. Na primeira posição do array estão as 32 imagens e na segunda suas classes/rótulos respectivamente. Com isso temos carregados 322 *batches* totalizando em 10298 imagens, 321 *batch* com 32 imagens cada e 1 *batch* com 26 imagens.

Seguindo o formato de separação do *dataset* e a quantidade de *batches*, os dados foram separados para seguir com o processamento. Para isso foram separados 70% dos *batches* para treinamento, resultando em um total de 225 *batches* com 7200 imagens. Outros 20% para validação do treinamento, total de 64 *batches* com 2048 imagens. O restante de 10% para testes resultou em 32 *batches* com 1024 imagens. Totalizando 321 *batches* com 10272 imagens selecionadas. Essa separação fez com que sobrasse algumas imagens em um *batch*. Para contornar isso podemos ignorar o *batch* ou colocar o restante das imagens em algum grupo de separação. Foi escolhido colocar esse *batch* restante junto com as imagens para função de testes. A Tabela 7 descreve a separação final após ajustes.

Tabela 7 – Separação do *dataset* para cada funcionalidade.

Função	Qtd. Batches	Qtd. Imagens
Treinamento	225	7200
Validação	64	2048
Testes	33 (32+1)	1050 (1024+26)

Fonte: Elaborado pelo autor.

Com os dados devidamente separados, foi dado início ao desenvolvimento de um protótipo para classificação. A ferramenta *Keras* ajudou nesse processo. Utilizando da classe '*Sequential*', o *Keras* possibilita a criação de um modelo que segue de forma sequencial entre as camadas ocultas. Como o nome sugere, a classe recebe os dados de entrada e segue em ordem sequencial entre as camadas até que os dados cheguem ao final do modelo. Esta classe ajuda na criação de um modelo, que é formado com camadas de informações ou dados, podendo ter múltiplas camadas interligadas.

Partindo dessa classe, podemos utilizar algumas camadas importantes para filtragem e extração de características das imagens. As camadas utilizadas são *Conv2D*, *MaxPooling2D*, *Flatten* e *Dense*.

A camada *Conv2D* é uma camada convolucional que aplica um filtro na imagem de entrada. Esse filtro tem tamanho 2D (X,Y) e percorre os pixels da imagem aplicando uma função

de ativação. A partir do resultado dessa função de ativação é gerado uma nova imagem como saída. A Figura 8 representa o processo convolucional de filtragem dessa camada. Geralmente a função de filtragem é um somatório da área do filtro resultando em um único pixel na nova imagem.

Conv2D é dependente de parâmetros para o processamento adequado. Quantos filtros, tamanho de cada filtro e quantidade de pixel que irá se mover, são os principais parâmetros. O primeiro parâmetro é a quantidade de filtros que serão utilizados para extrair características, deve ser passado como um número inteiro. O segundo parâmetro é o tamanho do filtro, nele definimos um retângulo que deve ser passado em formato de tupla, por exemplo (3,3). Nesse contexto uma tupla equivale a uma matriz (X,Y), onde o primeiro número equivale ao X e o segundo ao Y. E como terceiro parâmetro temos a função de ativação. Essa função será a responsável por ler os dados da área aplicada pelo filtro e processar o resultado.

A camada *MaxPooling2D* é uma operação de agrupamento que calcula o valor máximo para uma área definida de uma imagem e o utiliza para criar uma nova imagem com recursos reduzidos. Essa funcionalidade captura a característica mais notável da área definida, deixando o modelo capaz de detectar a classe à qual a imagem pertence independente do local em que a característica estiver. Existem alguns parâmetros de tamanho da área de agrupamento e tamanho do pulo que a área irá fazer para processar. Os valores padrão para essa camada são (2,2) para o tamanho e (2,2) para o salto em X e Y que irá fazer.

A camada *Flatten* faz a conversão da imagem em uma matriz com uma dimensão. A saída das camadas convolucionais é achatada para criar um único vetor. Exemplo desse processo é uma matriz 3x3 de entrada, que após ser processada pela camada *Flatten* é transformada em uma matriz 1x9.

A camada *Dense* é a *hidden layer* de um modelo. É o processo usado para criar camadas totalmente conectadas, nas quais cada saída depende de cada entrada. A camada alimenta todas as saídas da camada anterior para todos os seus neurônios, cada neurônio fornecendo uma saída para a próxima camada. Também é usada como camada final para apresentar o resultado de uma predição.

O modelo criado segue a seguinte estrutura. Iniciando pelas camadas convolucionais de filtragem, foi criado 4 grupos de camadas convolucionais. Cada grupo é composto por uma camada *Conv2D* e uma *MaxPooling2D*. Após as camadas convolucionais, a saída é achatada pela camada *Flatten* para ser processada pela camada *Dense*. Por fim mais uma camada *Dense* que fará a saída da predição para cada classe. A Tabela 8 descreve a estrutura do modelo criado.

Com as camadas criadas, basta definir qual serão as funções de otimização, as métricas e função de perda para usar na predição.

A função de otimização escolhida para o treinamento foi a *Adam*, pois aproveita dos métodos de taxas de aprendizado adaptativo para encontrar taxas de aprendizado individuais para cada parâmetro. *Adam* é popular entre os modelos de *Deep Learning* por sua robustez e resultados positivos. Como padrão *Adam* tem a taxa de aprendizado 0.001 e foi mantido para

Tabela 8 – Camadas Criadas para o Modelo de Classificação.

Camada	Saída	Parâmetros
Conv2D	(None, 198, 198, 8)	224
MaxPooling2D	(None, 99, 99, 8)	0
Conv2D	(None, 97, 97, 16)	1168
MaxPooling2D	(None, 48, 48, 16)	0
Conv2D	(None, 46, 46, 32)	4640
MaxPooling2D	(None, 23, 23, 32)	0
Conv2D	(None, 21, 21, 16)	4624
MaxPooling2D	(None, 10, 10, 16)	0
Flatten	(None, 1600)	0
Dense	(None, 360)	576360
Dense	(None, 5)	1805

Fonte: Elaborado pelo autor.

o treinamento. Para função de perda escolhida foi *sparse_categorical_crossentropy*, que é a função de perda ideal para problemas de classificação multiclases. Ela reduz o resultado para valores inteiros o que reduz o consumo de memória.

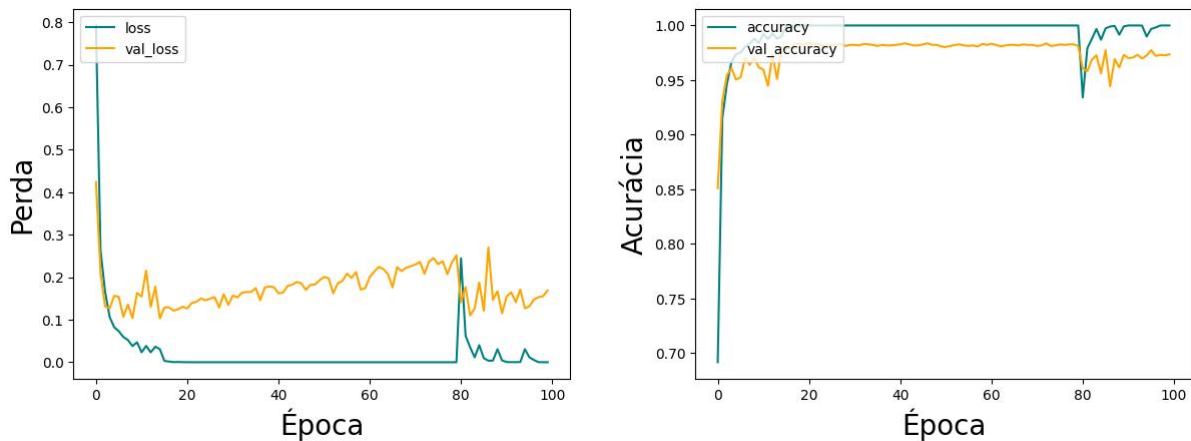
4.4 ANÁLISE DOS RESULTADOS

Com o otimizador compilado e o modelo pronto, a próxima etapa é a de treinamento. De início foi efetuado um treinamento de teste para verificar configuração do modelo e resultados preliminares. Para isso foi utilizado a GPU *NVIDIA RTX 3070* com 100 épocas. O tempo de cada treinamento para cada época variou entre 10 e 20 segundos, resultando em um tempo médio de 15 segundos, totalizando aproximadamente 30 minutos.

Após o treinamento de teste foi analisado os resultados preliminares obtidos para Perda e Acurácia de Treinamento e Validação. A Figura 13 demonstra um teste de treinamento por 100 épocas. Podemos ver que ao decorrer do tempo a perda de treinamento e de validação estavam progredindo a um valor ideal. Juntamente, temos a acurácia de treinamento e de validação, que também estavam convergindo ao valor ideal. Verificamos que a perda e a acurácia no treinamento atingiram valor máximo, enquanto perda e acurácia de validação regrediam. O que representa uma falha no aprendizado, já que os dados de validação não estavam mantendo a convergência, podendo representar *overfitting*.

Overfitting é o resultado de uma classificação errada feita pelo modelo na validação dos dados, enquanto classifica corretamente para o treinamento. Como visto, o modelo chegou ao valor ideal para acurácia dos dados de treinamento, enquanto para os dados de validação o modelo estava regredindo. Podemos ver essa característica após a época 20.

Figura 13 – Perda e Acurácia por 100 Épocas



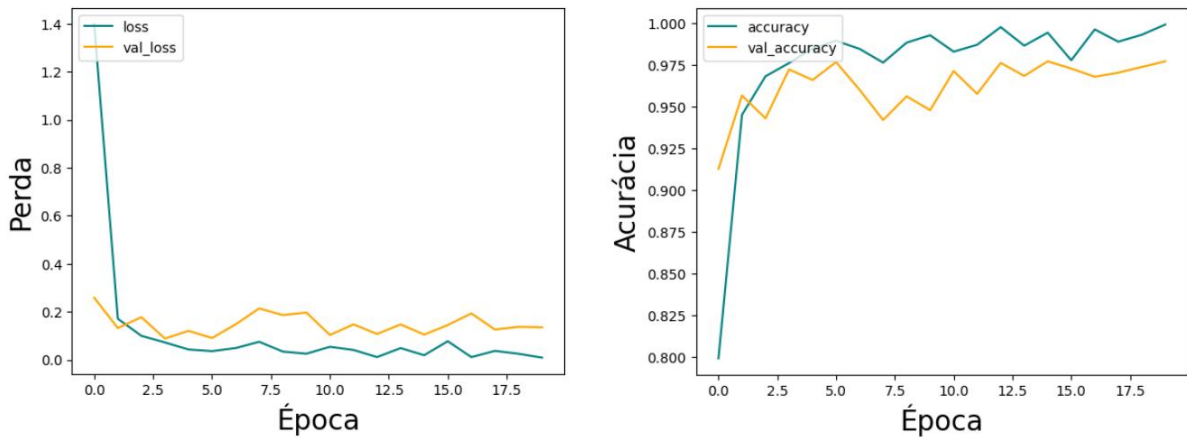
(a) Perda ao decorrer das épocas.

(b) Acurácia ao decorrer das épocas.

Fonte: Elaborado pelo autor.

Com base nesse teste, foi assumido 20 épocas como um número suficiente para treinamento. Foram usados 225 *batches* com 7200 imagens para o novo treinamento. Cada época do treinamento durou em média de 15 segundos utilizando a GPU *NVIDIA RTX 3070*, oque totalizou uma duração entre 5 e 6 minutos. A Figura 14 demonstra a perda e a acurácia ao decorrer das épocas de treinamento. Nela podemos ver que o modelo não sofreu por *overfitting*, pois os valores de treinamento e validação progrediam na mesma proporção.

Figura 14 – Perda e Acurácia por 20 Épocas



(a) Perda ao decorrer das épocas.

(b) Acurácia ao decorrer das épocas.

Fonte: Elaborado pelo autor.

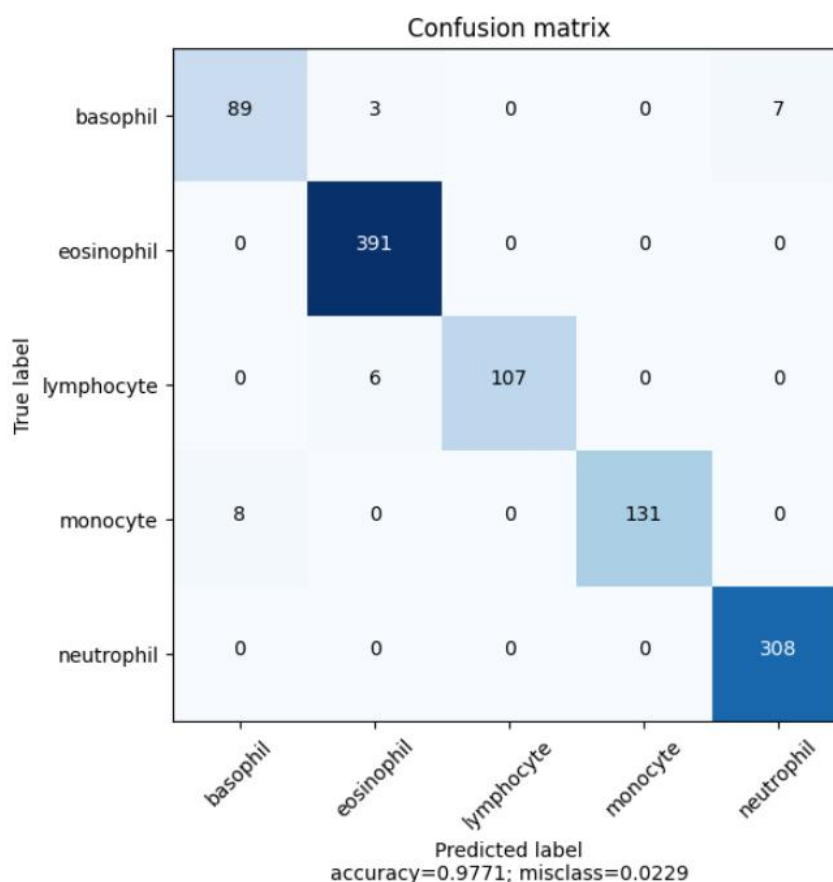
A Matriz de Confusão foi utilizada como métrica de avaliação do modelo. Essa métrica é uma forma de mapear as predições para as classes originais às quais os dados pertencem, através de uma tabela que relaciona o número de verdadeiros positivos (VP), verdadeiros negativos

(VN), falsos positivos (FP) e falsos negativos (FN). Para problemas de múltiplas classes, essa métrica afere relações para cada classe, onde a classe atual é vista como positivo e demais classes como negativo.

Nesse contexto, os casos de VP são os casos onde ocorreu uma classificação correta da classe em que o dado pertence. Os casos de VN são os que o modelo previu um dado de outra classe qualquer. FP são previsões erradas da classe do dado. Por fim os FN são os casos em que o modelo previu ser da classe, porém não é correto.

A Figura 15 demonstra a Matriz de Confusão para os dados de testes. Juntamente com a Matriz de Confusão, a Figura 16 apresenta uma tabela de Métricas de Classificação Multiclasses. Nela é possível confirmar dados de Precisão do Modelo (*Precision*), Proporção de Verdadeiros-Positivos (*Recall*) e a média entre eles. Na tabela de métricas podemos conferir que a *Precision* ficou definida em 0.97, o *Recall* ficou em 0.96, o que resulta em uma média final de 0.97 por arredondamento.

Figura 15 – Resultado Matriz de Confusão.



Fonte: Elaborado pelo autor.

Figura 16 – Métrica Multiclasses.

	precision	recall	f1-score	support
basophil	0.92	0.90	0.91	99
eosinophil	0.98	1.00	0.99	391
lymphocyte	1.00	0.95	0.97	113
monocyte	1.00	0.94	0.97	139
neutrophil	0.98	1.00	0.99	308
accuracy			0.98	1050
macro avg	0.97	0.96	0.97	1050
weighted avg	0.98	0.98	0.98	1050

Fonte: Elaborado pelo autor.

Nesse teste podemos ver a quantidade de imagens que estavam presente no *dataset* de testes. Como o *dataset* escolhido não tinha uma distribuição igualitária de imagens por classe, podemos perceber que a tabela apresenta uma métrica com pesos baseados na quantidade de imagens por classe. *Macro Avg* é a média da métrica sem peso por classe, enquanto *Weighted Avg* é a média da métrica com peso por classe. Essa informação é útil para confirmar que mesmo tendo um *dataset* não igualmente distribuído, o modelo conseguiu aprender com qualidade excepcional.

Satisfeito com o resultado e confirmando que o modelo não sofreu *overfitting*, o modelo foi salvo para utilização no protótipo de contagem e classificação desenvolvido por CRUZ (2).

4.5 EXPERIMENTO

Para o experimento foi utilizado o protótipo de visualização de detecção desenvolvido por CRUZ (2). Nele é carregado uma imagem de amostra de sangue, que é utilizada para detecção de Células Brancas (WBC), Células Vermelhas (RBC) e Plaquetas (Platelets). A Figura 17 apresenta a tela inicial com uma imagem de amostra de sangue e suas detecções com os títulos.

Foi necessário fazer pequenas alterações no código do protótipo para que o modelo de classificação pudesse ser encaixado no fluxo de detecção. Inicialmente foi feito o carregamento do modelo salvo anteriormente, utilizando uma ferramenta de carregamento do *Keras*.

Próximo passo é obter a imagem da célula branca detectada. Nesse passo foram importantes alguns processos como, validação da taxa de confiança do modelo de detecção de objetos para que não classificamos algo que não é uma célula branca detectada. Outro ponto importante é a caixa de detecção da célula. Ela será utilizada como molde para extração da célula branca da imagem original. Dessa maneira teremos uma imagem de célula branca no tamanho padronizado pelo modelo, que é 200x200 px. Caso a caixa de detecção não for tamanho suficiente, foi ajustado adicionado uma área 10% a mais para o tamanho da caixa, dessa forma a

Figura 17 – Protótipo para Detecção de Células.

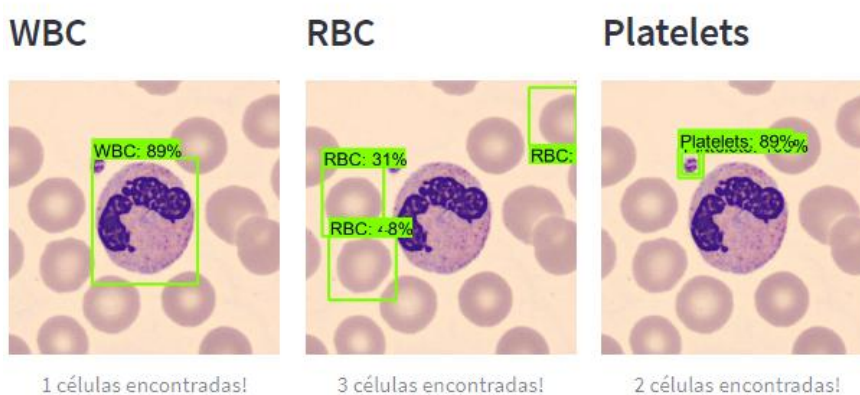
Object Detection e CNN Aplicado em Células Sanguíneas

O modelo Object Detection para detecção foi treinado utilizando o [BCCD Dataset](#), para fins de teste deve-se utilizar imagens dessa fonte.

O modelo CNN de Classificação foi treinado utilizando o [PBC Dataset](#).

Escolha uma imagem:

BNE_12343.jpg



1 células encontradas!

3 células encontradas!

2 células encontradas!

Fonte: Reproduzido de (2)

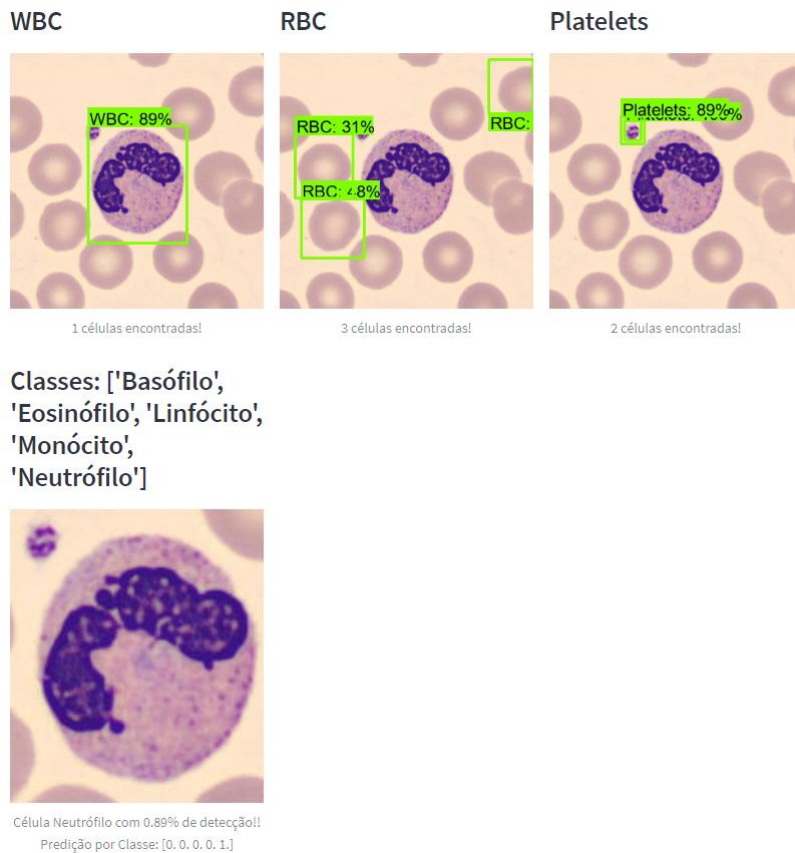
imagem extraída terá uma chance maior de conter toda a célula branca para classificação. Caso a caixa ajustada ultrapasse do limite da imagem, será atribuído o limite e ajustada para o formato desejado.

Também foi ajustado o protótipo para apresentação das classes detectadas na imagem. Foi necessário ajustar o layout de apresentação das células detectadas para ter espaço para as células classificadas. Na Figura 18 podemos ver como ficou o layout após ajustes.

Na coluna da Célula Branca, WBC, foi adicionado uma nova linha para cada célula detectada. Em cada linha terá as seguintes informações sobre a célula: Tipo classificado com maior confiança, a porcentagem da detecção da célula branca, e a lista de confiança por tipo de célula. A nova linha para as classes tem a definição da ordem dos tipos de célula, neste caso são elas: 1-Basófilo; 2-Eosinófilo; 3-Linfócito; 4-Monócito; e 5-Neutrófilo. É importante ter essa informação por que nem sempre o modelo irá retornar certeza na classificação. Desse modo o usuário terá uma opção para verificar se a classificação está correta.

Na Figura 19 podemos verificar as informações destacadas. Como Tipo Classificado foi Neutrófilo. Ela corresponde a célula que teve 78% de taxa de confiança na detecção. E a Predição por Classe apresenta a seguinte estrutura: [1. 22. 0. 6. 70.]. Essa predição por classe é a confiança na classificação, que para esse caso o modelo retornou 1% para Basófilo, 22% para Eosinófilo, 2% para Linfócito, 6% para Monócito e 70% para Neutrófilo. Com isso,

Figura 18 – Protótipo para Classificação das Células Brancas.



Fonte: Elaborado pelo autor.

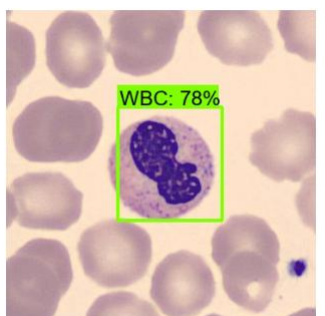
assumimos o maior número entre eles, como tipo classificado.

Com o protótipo para testes, foi verificado se o modelo continua classificando corretamente com trocar base de dados diferente da treinada. O *dataset*, *BCCD_Dataset* (28), utilizado por (2), foi utilizado para verificar se o modelo consegue generalizar com imagens fora do padrão treinado.

Na primeira imagem aleatória selecionada, na Figura 20, podemos conferir a classificação Eosinófilo, que parece estar correta. Como não temos o rótulo correto, estamos verificando visualmente. As características mais marcantes da células Eosinófilo, é que o núcleo é composto por um granulado de cor avermelhada. Com base nisso podemos afirmar que essa classificação está correta.

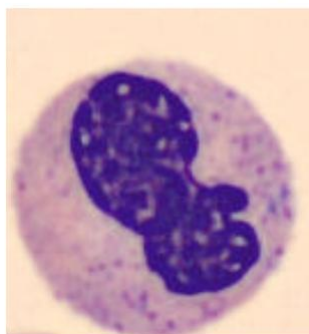
Figura 19 – Coluna de Classificação das Células Brancas no Protótipo.

WBC



1 células encontradas!

Classes: ['Basófilo',
'Eosinófilo', 'Linfócito',
'Monócito', 'Neutrófilo']



Célula Neutrófilo com 0,78% de detecção!! Predição
por Classe: [1. 22. 0. 6. 70.]

Fonte: Elaborado pelo autor.

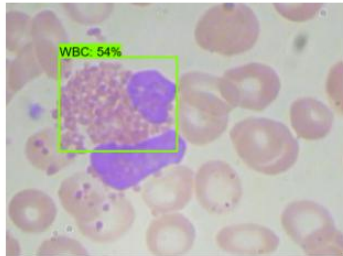
Segunda imagem foi manualmente selecionada para pegarmos o caso onde não há detecção da célula branca com taxa aceitável. Na Figura [21](#) podemos verificar como que o protótipo se comporta.

Figura 20 – Classificação Eosinófilo.

Insira o link direto da imagem:

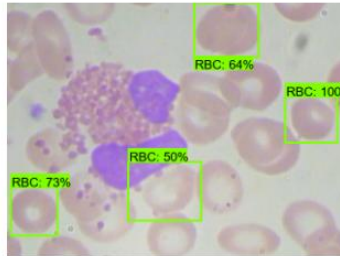
https://raw.githubusercontent.com/Shenggan/BCCD_Dataset/master/BCCD/JPEGImages/BloodImage_00344.jpg

WBC



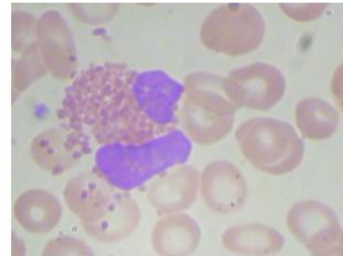
1 células encontradas!

RBC



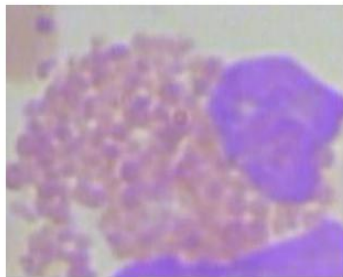
4 células encontradas!

Platelets



0 células encontradas!

**Classes: ['Basófilo',
'Eosinófilo', 'Linfócito',
'Monócito', 'Neutrófilo']**

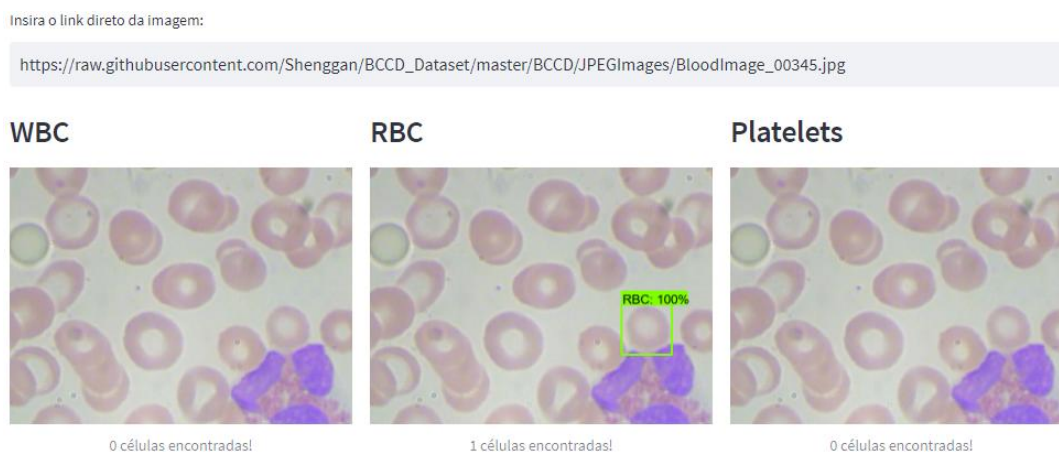


Célula Eosinófilo com 0.54% de detecção!! Predição por Classe: [0. 96. 0. 3. 0.]

Fonte: Elaborado pelo autor.

Próxima imagem também foi manualmente selecionada. Com ela podemos verificar que quando há detecção de mais de uma célula por imagem, a classificação é feita em cada detecção, mesmo que uma delas esteja apenas com a caixa de detecção menor. Na Figura 22 vemos duas imagens classificadas com tipos diferentes, por se tratar de células iguais com extração diferente. Nesse caso a primeira imagem com taxa de 0.73% está correta, a segunda com 0.50% de taxa está classificando errada por se tratar de uma imagem deslocada.

Figura 21 – Sem classificação.

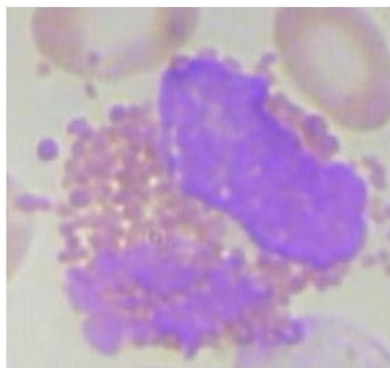


Fonte: Elaborado pelo autor.

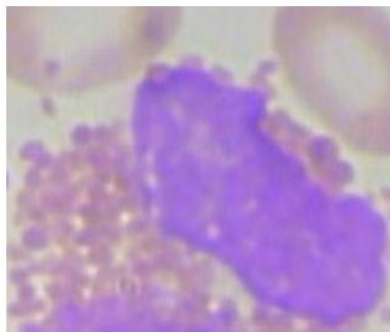
Por fim temos um caso onde o modelo de classificação retornou menos de 50% de confiança na predição. Na Figura 23 podemos verificar que na Predição por Classe o modelo retornou 47% de confiança para Eosinófilo, 20% para Monócito e 34% para Neutrófilo. Como assumimos a maior confiança, esse caso o modelo classificou como Eosinófilo erroneamente.

O que podemos verificar com o experimento no protótipo é que a generalização não tem alta precisão na classificação quando uma imagem não está no padrão do treino. A imagem do *dataset* BCCD não estão nítidas o suficiente para serem comparados as imagens com que o modelo foi treinado. Dessa forma a classificação fica dependente de uma característica muito predominante, como a coloração da célula, fazendo com que a maioria se encaixe como Monócito, que contém uma grande quantidade de cor predominante.

Figura 22 – Classificação Dupla.



Célula Eosinófilo com 0.73% de detecção!! Predição por Classe: [0. 99. 0. 1. 0.]



Célula Monócito com 0.5% de detecção!! Predição por Classe: [0. 11. 2. 85. 2.]

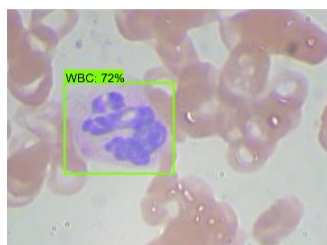
Fonte: Elaborado pelo autor.

Figura 23 – Predição para cada Classe.

Insira o link direto da imagem:

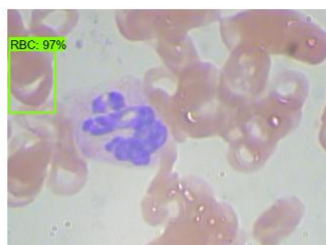
https://raw.githubusercontent.com/Shenggan/BCCD_Dataset/master/BCCD/JPEGImages/BloodImage_00004.jpg

WBC



1 células encontradas!

RBC



1 células encontradas!

Platelets



0 células encontradas!

Classes: ['Basófilo',
'Eosinófilo', 'Linfócito',
'Monócito', 'Neutrófilo']



Célula Eosinófilo com 0.72% de detecção!! Predição
por Classe: [0. 47. 0. 20. 34.]

Fonte: Elaborado pelo autor.

5 CONCLUSÕES

O objetivo desse trabalho era treinar um modelo computacional a ponto de ser capaz de classificar as células brancas em uma imagem de amostra de sangue. Para isso abordamos os conhecimentos básicos sobre sangue e suas células, *deep learning* e suas técnicas de Rede Neurais Artificiais. Dentre as células destacadas, as mais importantes são as células brancas, que são os objetos de pesquisa. Classificá-las em uma imagem de amostra de sangue é essencial para a elaboração de um hemograma completo. Vimos que o hemograma completo é o exame corriqueiro em laboratórios, por ser capaz de avaliar o estado de saúde de um paciente e diagnosticar doenças de forma não invasiva. Esse exame ocupa muito tempo do profissional, que se fosse reduzido via automação, poderia estar focando em outras tarefas. Com essa perspectiva, foi criado um modelo para automatizar a classificação de células brancas. Desse modo o profissional poderá agilizar o processo de produção de hemograma completo.

Entendemos o processo de produção de um hemograma completo, com suas partes e tabela de valores. Desbravamos conceitos de Redes Neurais Convolucionais para classificação de imagens, como ela é treinada, como suas camadas funcionam e ferramentas que nos auxiliam no dia a dia para modelagem.

Dando sequência ao trabalho de CRUZ (2), o modelo de classificação se encaixa no protótipo de detecção de células já desenvolvida. Vimos o funcionamento da detecção e a apresentação de seus resultados. Alteramos o funcionamento para que cada célula branca detectada passe pelo modelo de classificação. E para apresentar o resultado da classificação foi adicionado o tipo da célula em cada detecção, apresentando visualmente.

A combinação do trabalho de detecção com o modelo de classificação é de grande utilidade, pois permite que o profissional tire conclusões sobre o estado de saúde do paciente. Com a detecção, contagem e classificação das células é possível calcular os valores necessários para elaboração de um hemograma completo.

Como os trabalhos unificados foram treinados com bases de dados diferentes, um apresenta resultados ineficiente quando aplicado aos dados do outro. Quando aplicado detecção nos dados de classificação, muitas vezes não detecta as células brancas corretamente, chegando até a duplicar a detecção. Quando aplicado classificação nos dados de detecção, a diferença entre qualidade e coloração faz com que o modelo erre nas predições.

Com a utilização de um *dataset* com imagens no mesmo padrão de coloração e qualidade, seria possível obter um resultado melhor na detecção e classificação dos elementos da amostra de sangue. Se o *dataset* utilizado para classificação tivesse dados relacionados a posição de cada célula, poderia ser utilizado no treinamento para detecção, melhorando o resultado da classificação. Com essa melhora nos resultados, poderíamos alcançar 97% de precisão, conforme o modelo de classificação obteve.

Para pesquisas futuras, é possível utilizar os conceitos e modelos deste trabalho para melhorar a predição com dados fora do padrão, utilizando um modelo pré treinado via *transfer*

learning. Também está aberto a elaboração de um resumo de dados para elaboração de hemograma completo, dentre eles, a estimativa de quantidade de células por volumetria de sangue. Os demais dados também podem ser extraídos, dessa forma o profissional teria as informações necessárias para elaboração de um hemograma completo e tomada de decisão, utilizando a ferramenta em todo o processo de contagem e classificação das células.

REFERÊNCIAS

- 1 KRITTANAWONG, Chayakrit et al. Deep learning for cardiovascular medicine: a practical primer. **European Heart Journal**, 2019. ISSN 0195-668X. DOI: [10.1093/eurheartj/ehz056](https://doi.org/10.1093/eurheartj/ehz056). eprint: <https://academic.oup.com/eurheartj/article-pdf/40/25/2058/28882900/ehz056.pdf>. Disponível em: <https://doi.org/10.1093/eurheartj/ehz056>.
- 2 CRUZ, ANTHONY. APLICANDO DEEP LEARNING PARA DETECÇÃO DE CÉLULAS EM AMOSTRAS DE SANGUE. **INSTITUTO FEDERAL DE SANTA CATARINA**, 2021. Disponível em: <https://repositorio.ifsc.edu.br/handle/123456789/2501>.
- 3 RAVÌ, Daniele et al. Deep Learning for Health Informatics. **IEEE Journal of Biomedical and Health Informatics**, 2017. DOI: [10.1109/JBHI.2016.2636665](https://doi.org/10.1109/JBHI.2016.2636665).
- 4 PFEIL, Juliane et al. Examination of blood samples using deep learning and mobile microscopy. **BMC Bioinformatics**, 2022. ISSN 14712105. Disponível em: <https://search.ebscohost.com/login.aspx?direct=true&db=aph&AN=155183533&site=ehost-live>.
- 5 DREWE-BOSS, Philipp et al. Deep learning for prediction of population health costs. **BMC Medical Informatics Decision Making**, 2022. ISSN 14726947. Disponível em: <https://search.ebscohost.com/login.aspx?direct=true&db=aph&AN=157024723&site=ehost-live>.
- 6 TREVISOL, Rodolfo. **White Blood Cell Classification Using CNN**. [S.l.: s.n.], 2023. Disponível em: <https://github.com/rodolfojr/white-blood-cell-classification>.
- 7 CELKAN, Tülin Tiraje. What does a hemogram say to us? en. **Türk Pediatri Ars**, Turkey, v. 55, n. 2, p. 103–116, jun. 2020. DOI: [10.14744/TurkPediatriArs.2019.76301](https://doi.org/10.14744/TurkPediatriArs.2019.76301).
- 8 LOKWANI, D. P. The ABC of CBC: Interpretation of Complete Blood Count and Histograms. In. DOI: [10.5005/jp/books/11929](https://doi.org/10.5005/jp/books/11929).
- 9 VIVAS, W. L. P. Manual Prático de Hematologia, 2017.
- 10 PNCQ. Valores De Referência Hematológicos para Adultos e Crianças. In. Disponível em: <https://pncq.org.br/wp-content/uploads/2021/04/VRH2020.pdf>.
- 11 KHAN, Siraj et al. A Review on Traditional Machine Learning and Deep Learning Models for WBCs Classification in Blood Smear Images. **IEEE Access**, 2021. DOI: [10.1109/ACCESS.2020.3048172](https://doi.org/10.1109/ACCESS.2020.3048172).

- 12 MAHESH, Batta. Machine Learning Algorithms - A Review. In. DOI: [10.21275/ART20203995](https://doi.org/10.21275/ART20203995). Disponível em: https://www.researchgate.net/publication/344717762_Machine_Learning_Algorithms_-_A_Review>.
- 13 SHRESTHA, Ajay; MAHMOOD, Ausif. Review of Deep Learning Algorithms and Architectures. **IEEE Access**, 2019. DOI: [10.1109/ACCESS.2019.2912200](https://doi.org/10.1109/ACCESS.2019.2912200).
- 14 TIWARI, Prayag et al. Detection of subtype blood cells using deep learning. **Cognitive Systems Research**, 2018. ISSN 1389-0417. DOI: <https://doi.org/10.1016/j.cogsys.2018.08.022>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1389041718303760>>.
- 15 MACAWILE, Merl James et al. White blood cell classification and counting using convolutional neural network. In: p. 259–263. DOI: [10.1109/ICCRE.2018.8376476](https://doi.org/10.1109/ICCRE.2018.8376476).
- 16 MARTÍN ABADI et al. **TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems**. [S.l.: s.n.], 2015. Software available from tensorflow.org. Disponível em: <https://www.tensorflow.org/>>.
- 17 CHOLLET, François et al. **Keras**. [S.l.: s.n.], 2015. <https://keras.io>.
- 18 RIDOY, Md. Alif Rahman; ISLAM, Md. Rabiul. An Automated Approach to White Blood Cell Classification Using a Lightweight Convolutional Neural Network. In: 2020 2nd International Conference on Advanced Information and Communication Technology (ICAICT). [S.l.: s.n.], 2020. DOI: [10.1109/ICAICT51780.2020.9333512](https://doi.org/10.1109/ICAICT51780.2020.9333512).
- 19 SIDDIQUE, Md. Abu Ismail; AZIZ, Abu Zahid Bin; MATIN, Abdul. An improved deep learning based classification of human white blood cell images. In: 2020 11th International Conference on Electrical and Computer Engineering (ICECE). [S.l.: s.n.], 2020. DOI: [10.1109/ICECE51571.2020.9393156](https://doi.org/10.1109/ICECE51571.2020.9393156).
- 20 SHINDE, Srushti et al. Analysis of WBC, RBC, Platelets Using Deep Learning. In: 2021 IEEE Pune Section International Conference (PuneCon). [S.l.: s.n.], 2021. DOI: [10.1109/PuneCon52575.2021.9686524](https://doi.org/10.1109/PuneCon52575.2021.9686524).
- 21 KOUSALYA, K. et al. Comparative analysis of White Blood Cells Classification using Deep Learning Architectures. In: 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC). [S.l.: s.n.], 2021. DOI: [10.1109/ICOSEC51865.2021.9591771](https://doi.org/10.1109/ICOSEC51865.2021.9591771).
- 22 HUSSAIN, Muhammad Abbas et al. Leukocytes Segmentation and Classification in Digital Microscopic Images. In: 2021 4th International Conference on Computing Information Sciences (ICCIS). [S.l.: s.n.], 2021. DOI: [10.1109/ICCIS54243.2021.9676191](https://doi.org/10.1109/ICCIS54243.2021.9676191).

- 23 NAVYA, K.T; PRASAD, Keerthana; SINGH, Brij Mohan Kumar. Classification of blood cells into white blood cells and red blood cells from blood smear images using machine learning techniques. In: 2021 2nd Global Conference for Advancement in Technology (GCAT). [S.l.: s.n.], 2021. DOI: [10.1109/GCAT52182.2021.9587524](https://doi.org/10.1109/GCAT52182.2021.9587524).
- 24 IQBAL, Ashik et al. Towards Efficient Segmentation and Classification of White Blood Cell Cancer Using Deep Learning. In: 2021 Emerging Technology in Computing, Communication and Electronics (ETCCE). [S.l.: s.n.], 2021. DOI: [10.1109/ETCCE54784.2021.9689839](https://doi.org/10.1109/ETCCE54784.2021.9689839).
- 25 ACEVEDO, Andrea et al. A dataset of microscopic peripheral blood cell images for development of automatic recognition systems. en. **Data Brief**, Netherlands, v. 30, p. 105474, abr. 2020. Disponível em: <https://data.mendeley.com/datasets/snkd93bnjr/1>.
- 26 NVIDIA; VINGELMANN, Péter; FITZEK, Frank H.P. **CUDA, release: 11.2**. [S.l.: s.n.], 2020. Disponível em: <https://developer.nvidia.com/cuda-toolkit>.
- 27 _____. **cuDNN, release: 8**. [S.l.: s.n.], 2020. Disponível em: <https://developer.nvidia.com/cudnn>.
- 28 SHENGGAN. **BCCD (Blood Cell Count and Detection) Dataset is a small-scale dataset for blood cells detection**. [S.l.: s.n.], 2019. Disponível em: https://github.com/Shenggan/BCCD_Dataset.

unsrc[heading=abnt]