

---

# **BioWebVis - Ambiente web para citemorfometria utilizando imagens 3D**

---

**Eduardo Henrique Silva**



UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
FACULDADE DE COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



**Eduardo Henrique Silva**

**BioWebVis - Ambiente web para  
citemorfometria utilizando imagens 3D**

Dissertação de mestrado apresentada ao Programa de Pós-graduação da Faculdade de Computação da Universidade Federal de Uberlândia como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Área de concentração: Ciência da Computação

Orientador: Bruno Augusto Nassif Travençolo  
Coorientador: Jefferson Rodrigo de Souza

Uberlândia  
2017

Dados Internacionais de Catalogação na Publicação (CIP)  
Sistema de Bibliotecas da UFU, MG, Brasil.

---

S586                   Silva, Eduardo Henrique, 1990-  
2017                   BioWebVis - ambiente web para citomorfometria utilizando imagens  
                         3D / Eduardo Henrique Silva. - 2017.  
                         95 f. : il.

Orientador: Bruno Augusto Nassif Travençolo.  
Coorientador: Jefferson Rodrigues de Souza.  
Dissertação (mestrado) - Universidade Federal de Uberlândia,  
Programa de Pós-Graduação em Ciência da Computação.  
Inclui bibliografia.

1. Computação - Teses. 2. Desenvolvimento de software - Teses. 3.  
Análise - Software - Teses. 4. Armazenamento de dados - Teses. I.  
Travençolo, Bruno Augusto Nassif. II. Souza, Jefferson Rodrigues de.  
III. Universidade Federal de Uberlândia. Programa de Pós-Graduação em  
Ciência da Computação. IV. Título.

*Para minha família, amigos que de alguma forma contribuíram e aos professores que foram importantes nessa trajetória.*



---

## Agradecimentos

Agradeço primeiramente a Deus, por me acompanhar durante essa trajetória, principalmente durante as viagens.

Aos meus pais Ilzidete e Vanderlei e ao meu irmão Wanderley Júnior pelo incentivo, ensinamentos e por me encorajar a não desistir.

A minha esposa Francielle que esteve ao meu lado o tempo todo me apoiando, pela paciência em meus momentos de ausência e principalmente por estar comigo nos momentos difíceis.

Ao Fernando Corrêa, por todo apoio e principalmente por ser a pessoa que me incentivou a iniciar a pós-graduação.

Aos professores Bruno Travençolo e Jefferson Rodrigo, pela orientação, apoio e ensinamentos ao longo do desenvolvimento deste projeto e ao professor Marcelo Beletti, pelo apoio e colaboração durante o desenvolvimento deste projeto.

Aos professores da Faculdade de Computação da Universidade Federal de Uberlândia, que construíram com a minha formação pessoal e profissional.



*“A vida é frágil. Nós não temos um amanhã garantido,  
então dê a ele tudo o que você tiver.”*  
*(Tim Cook)*



---

## Resumo

Os softwares de bioimagens são utilizados para a análise de imagens microscópicas e auxiliam os usuários em suas tomadas de decisões. Grande parte dos softwares de bioimagens são disponibilizados como ferramentas locais, necessitando de instalação. A presente dissertação visa apresentar o desenvolvimento de um software de bioimagem na nuvem, denominado BioWebVis. O BioWebVis pode ser acessado em um navegador web utilizando a internet. Pretende-se com esse software auxiliar patologistas, biólogos e outros especialistas na tomada de decisão, minimizando a subjetividade de suas avaliações. No desenvolvimento foram realizados estudos nos softwares de bioimagens livres (gratuitos), o qual foram identificadas as características positivas e negativas, permitindo a escolha das tecnologias apropriadas para desenvolver o ambiente. O trabalho proveu um software de bioimagem na nuvem para análise citomorfométrica usando imagens tridimensionais. Para validar o ambiente foram realizados dois estudos de casos. Em cada estudo de caso, técnicas diferentes foram utilizadas para a validação das funcionalidades do ambiente. O primeiro, utilizou a citomorfometria para classificação da localização de padrões subcelulares em células HeLa. Esse estudo de caso forneceu resultados relevantes, pois utilizando o classificador Análise Discriminante Quadrática (*Quadratic Discriminant Analysis*) (QDA) com 5 atributos morfológicos atingiu-se a acurácia de 97,59% na classificação de padrões subcelulares. O segundo estudo de caso, utilizou a citomorfometria para a análise das alterações no padrão de heterocromatina nos cérebros de tartarugas da Amazônia. Utilizando o algoritmo *K-means* obteve-se o *V-measure* igual a 1, indicando um agrupamento perfeito. Estudos preliminares indicam que as células de heterocromatina sofreram uma diminuição de tamanho sobre dosagens maiores de herbicidas. O ambiente proposto foi capaz fornecer dados quantitativos para a análise citomorfométrica e também auxiliou na descoberta de padrões celulares.

**Palavras-chave:** Citomorfometria. Software de bioimagem. Visualização científica.



---

# Abstract

Bioimaging software is used to analyze microscopic images and assist users in their decision making. Most bioimaging software is available as local tools, requiring installation and, in some cases, configuration. The present dissertation aims to present the development of bioimaging software in the cloud, called BioWebVis. BioWebVis can be accessed in a web browser using the internet. This software is intended to assist pathologists, biologists and other specialists in decision making, minimizing the subjectivity of their evaluations. In the development, studies were carried out on free bioimaging software, which identified the positive and negative characteristics, allowing the selection of appropriate technologies to develop the environment. The work provided a bioimaging software in the cloud for cytomorphometric analysis using three-dimensional images.

Two case studies were carried out to validate the environment. In each case study, different techniques were used to validate the environmental functionalities. The first one used cytomorphometry to classify the location of subcellular patterns in HeLa cells. This case study provided relevant results, because using the Quadratic Discriminant Analysis (QDA) classifier with 5 morphological attributes made possible to reach an accuracy of 97.59% in the classification of subcellular patterns. The second case study used cytomorphometry to analyze changes in the heterochromatin pattern in Amazonian turtle brains. Using the K-means algorithm the V-measure was equal to 1, indicating a perfect grouping. Preliminary studies indicate that heterochromatin cells have suffered a decrease in size with larger dosages of herbicides. The proposed environment was able to provide quantitative data for cytomorphometric analysis and also aided in the discovery of cellular patterns.

**Keywords:** Cytomorphometry. Bioimaging software. Scientific visualization..



---

# Listas de ilustrações

Figura 1 – Processo de análise e visualização. . . . .	25
Figura 2 – Algoritmo Volume Ray-casting. . . . .	27
Figura 3 – <i>Marching Cubes</i> . . . . .	28
Figura 4 – Triangulações do <i>Marching Cubes</i> . . . . .	28
Figura 5 – Reconstrução 3D com o software BioImageXD. . . . .	29
Figura 6 – Representação de um polígono $P$ não convexo e seu <i>convex hull</i> $CH(P)$ . . . . .	31
Figura 7 – <i>Convex hull</i> de uma célula. . . . .	32
Figura 8 – Representação da superfície para inferir a curvatura Gaussiana no ponto $p$ . . . . .	33
Figura 9 – PCA. . . . .	36
Figura 10 – Aprendizado de máquina. . . . .	37
Figura 11 – BioImageXD. . . . .	42
Figura 12 – Visão geral do BioWebVis. . . . .	51
Figura 13 – Tela inicial do BioWebVis. . . . .	52
Figura 14 – Funcionalidades do BioWebVis para ViSC e Citomorfometria. . . . .	53
Figura 15 – Visualização de um objeto de interesse. . . . .	55
Figura 16 – Funcionalidades para análise dos atributos. . . . .	56
Figura 17 – Arquitetura do ambiente BioWebVis. . . . .	58
Figura 18 – Gráfico de desempenho das bibliotecas para manipulação do WebGL. . . . .	62
Figura 19 – Visualização da base de dados utilizando o software BioWebVis. . . . .	67
Figura 20 – <i>Pipeline</i> geral utilizado para análise dos padrões subcelulares. . . . .	68
Figura 21 – Visualização da isosuperfície da classe LAMP2 utilizando o BioWebVis. . . . .	69
Figura 22 – Visualização das isosuperfícies utilizando o BioWebVis. . . . .	70
Figura 23 – Análise de regressão entre os atributos de entropia e média da curva- tura. $\rho = -0,079$ . . . . .	72
Figura 24 – Análise de regressão entre os atributos de entropia da curvatura e es- fericidade. $\rho = -0,957$ . . . . .	72

Figura 25 – Visualização da base de dados após a segmentação utilizando o software BioWebVis. . . . .	79
Figura 26 – <i>Pipeline</i> geral utilizado para análise das alterações no padrão de heterocromatina nos cérebros de tartarugas. . . . .	80
Figura 27 – Visualização da amostra de células de tartaruga após a segmentação utilizando o software BioWebVis. . . . .	81
Figura 28 – Visualização heterocromatina de uma célula utilizando o software Bi- oWebVis. . . . .	82
Figura 29 – Análise de regressão entre os atributos da área da superfície e volume. $\rho = 0,99$ . . . . .	84

---

## **Lista de tabelas**

Tabela 1 – Medidas estatísticas sobre os coeficientes de curvatura. . . . .	34
Tabela 2 – Comparação da visualização multidimensional dos trabalhos relacionados. . . . .	45
Tabela 3 – Comparação dos algoritmos de processamento dos trabalhos relacionados. . . . .	46
Tabela 4 – Comparação dos algoritmos de análise quantitativa dos trabalhos relacionados. . . . .	47
Tabela 5 – Comparação das arquiteturas e outras funcionalidades com os trabalhos relacionados. . . . .	48
Tabela 6 – Tamanhos de diferentes tipos de arquivos para representação 3D. . . . .	61
Tabela 7 – Desempenho das bibliotecas de visualização com WebGL. . . . .	62
Tabela 8 – Resultado dos classificadores com os todos os 17 atributos. . . . .	71
Tabela 9 – Resultado dos classificadores com os 7 atributos selecionados com o <i>Extra Tree</i> . . . . .	71
Tabela 10 – Resultado dos classificadores com os 5 atributos selecionados com <i>Extra Tree</i> e análise de correlação. . . . .	73
Tabela 11 – Matriz de confusão. . . . .	73
Tabela 12 – Comparação dos resultados com outros trabalhos relacionados . . . . .	74
Tabela 13 – Resultado do agrupamento. . . . .	83



---

## **Lista de siglas**

**API** Interface de Programação de Aplicação (*Application Programming Interface*)

**Bisque** *Bio-Image Semantic Query User Environment*

**BioIMAX** *BioImage Mining, Analysis and eXploration*

**BPNN** *Back-Propagation Neural Network*

**CSS 3** Folhas de Estilo em Cascata 3 (*Cascading Style Sheets 3*)

**CSV** Valores Separados por Vírgulas (*Comma-separated values*)

**DVR** Renderização Volumétrica Direta (*Direct Volume Renderer*)

**DTA** Algoritmo do Teorema da Divergência (*Divergence theorem algorithm*)

**HTML 5** Linguagem de Marcação de Hipertexto (*HyperText Markup Language 5*)

**HTTP** Protocolo de Transferência de Hipertexto (*Hypertext Transfer Protocol*)

**IHC** Interação humano-computador

**K-NN** *K-Nearest Neighbors*

**MC** *Marching Cubes*

**ML** Aprendizado de Máquina (*Machine Learning*)

**NSI** Índice de Forma Normalizado (*Normalized Shape Index*)

**OMERO** *OME Remote Objects*

**PCA** Análise de Componentes Principais (*Principal Component Analysis*)

**ppb** partes por bilhão

**QDA** Análise Discriminante Quadrática (*Quadratic Discriminant Analysis*)

**RE** *Retículo Endoplasmático*

**SDA** Análise Discriminante Gradativa (*Stepwise Discriminant Analysis*)

**SF** Ajuste de Superfície (*Surface-Fitting*)

**SVM** Máquina de Vetor Suporte (*Support Vector Machine*)

**WebGL** Biblioteca de Gráficos Web (*Web Graphics Library*)

**VI** Visualização de Informação

**ViSC** Visualização Científica

**VTK** *Visualization Toolkit*

**XML** Linguagem de Marcação Extensível (*eXtensible Markup Language*)

**XTK** X Toolkit

---

# **Sumário**

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>21</b>
1.1	Motivação . . . . .	22
1.2	Objetivos . . . . .	23
1.2.1	Geral . . . . .	23
1.2.2	Específicos . . . . .	23
1.3	Contribuições . . . . .	23
1.4	Organização da Dissertação . . . . .	24
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA . . . . .</b>	<b>25</b>
2.1	Visualização de Informação e Científica . . . . .	25
2.1.1	Renderização Volumétrica Direta . . . . .	27
2.1.2	Isosuperfície . . . . .	27
2.2	Análise Citomorfométrica . . . . .	29
2.2.1	Área da Superfície . . . . .	30
2.2.2	Volume . . . . .	30
2.2.3	Convexidade . . . . .	31
2.2.4	Esfericidade . . . . .	32
2.2.5	Diâmetro Esférico Equivalente . . . . .	33
2.2.6	Curvatura . . . . .	33
2.3	Reconhecimento de Padrões . . . . .	34
2.3.1	Normalização . . . . .	35
2.3.2	Seleção de Atributos . . . . .	35
2.3.3	Aprendizado de Máquina . . . . .	36
<b>3</b>	<b>TRABALHOS RELACIONADOS . . . . .</b>	<b>41</b>
3.1	Softwares de bioimagens de execução local . . . . .	41
3.2	Softwares de bioimagens na nuvem . . . . .	43
3.3	Comparação com trabalhos relacionados . . . . .	44

<b>4</b>	<b>BIOWEBVIS - AMBIENTE WEB PARA CITOMORFOMETRIA UTILIZANDO IMAGENS 3D . . . . .</b>	<b>51</b>
<b>4.1</b>	<b>Visão Geral . . . . .</b>	<b>51</b>
4.1.1	Visualização 3D e Citomorfometria . . . . .	52
4.1.2	Análise e Aprendizado de Máquina . . . . .	56
<b>4.2</b>	<b>Arquitetura . . . . .</b>	<b>57</b>
<b>4.3</b>	<b>Visualização Usando WebGL . . . . .</b>	<b>60</b>
4.3.1	Bibliotecas WebGL . . . . .	60
4.3.2	Avaliação das Bibliotecas . . . . .	61
<b>5</b>	<b>USO DA CITOMORFOMETRIA PARA CLASSIFICAÇÃO DA LOCALIZAÇÃO DE PADRÕES SUBCELULARES EM ISOSUPERFÍCIES 3D . . . . .</b>	<b>65</b>
<b>5.1</b>	<b>Introdução . . . . .</b>	<b>65</b>
5.1.1	Trabalhos relacionados . . . . .	66
<b>5.2</b>	<b>Materiais e Métodos . . . . .</b>	<b>67</b>
5.2.1	Base de Dados . . . . .	67
5.2.2	Metodologia Proposta . . . . .	68
<b>5.3</b>	<b>Resultados . . . . .</b>	<b>69</b>
<b>5.4</b>	<b>Conclusão . . . . .</b>	<b>75</b>
<b>6</b>	<b>USO DA CITOMORFOMETRIA PARA A ANÁLISE DAS ALTERAÇÕES NO PADRÃO DE HETEROCROMATINA NOS CÉREBROS DE TARTARUGAS DA AMAZÔNIA . . . . .</b>	<b>77</b>
<b>6.1</b>	<b>Introdução . . . . .</b>	<b>77</b>
<b>6.2</b>	<b>Material e Métodos . . . . .</b>	<b>78</b>
6.2.1	Base de Dados . . . . .	78
6.2.2	Metodologia Proposta . . . . .	79
<b>6.3</b>	<b>Resultados . . . . .</b>	<b>82</b>
<b>6.4</b>	<b>Conclusão . . . . .</b>	<b>84</b>
<b>7</b>	<b>CONCLUSÃO . . . . .</b>	<b>87</b>
<b>7.1</b>	<b>Principais Contribuições . . . . .</b>	<b>88</b>
<b>7.2</b>	<b>Trabalhos Futuros . . . . .</b>	<b>88</b>
<b>7.3</b>	<b>Contribuições em Produção Bibliográfica . . . . .</b>	<b>88</b>
<b>REFERÊNCIAS . . . . .</b>		<b>91</b>

# CAPÍTULO 1

## Introdução

Imagens fornecidas por microscópios passaram por grandes avanços nos últimos anos devido à evolução das tecnologias ópticas e não ópticas de recuperação de imagens. Com esses avanços, métodos computacionais são necessários para auxiliar os cientistas nas tarefas de aquisição, armazenamento, processamento, análise e visualização das imagens. Todas essas tarefas são fundamentais para os cientistas, porém os métodos de análise e visualização são indispensáveis para obtenção de resultados e requerem uma diversidade de ferramentas a fim de extrair dados quantitativos (ELICEIRI et al., 2012).

O termo “software de bioimagem” surgiu para categorizar as aplicações que manipulam imagens microscópicas. Os métodos computacionais disponibilizados por esses softwares são uma importante ferramenta para diminuir a subjetividade e possibilitar que os biólogos concentrem-se na formulação de hipóteses e na concepção de experimentos, melhorando a reproduzibilidade e objetividade dos experimentos por meio de uma análise quantitativa (WALDEMARIN; BELETTI; COSTA, 2004) (MEIJERING et al., 2016).

Um estudo publicado na revista *Nature Methods* (CARDONA; TOMANCAK, 2012), apresentou que a usabilidade dos softwares de bioimagens tem sido um desafio e pode ser um fator limitante para o seu uso, pois cientistas de diferentes áreas, com conhecimento básico em computação, devem ser capazes de realizarem os seus experimentos no software disponibilizado. Características como a facilidade na instalação, suporte a várias plataformas, atualização automática e integração com outros softwares podem contribuir com a usabilidade. O estudo ainda ressalta que a disponibilização do software na nuvem evita muitos problemas de instalação e possibilita sua utilização remota, além de permitir o compartilhamento dos resultados com outros cientistas, quebrando barreiras geográficas. Segundo Eliceiri et al. (2012), disponibilizar um software de bioimagem em uma arquitetura de computação em nuvem permite aproveitar a escalabilidade, ou seja, o aumento do poder computacional de forma simplificada, fornecida por esse estilo arquitetural.

Assim, neste trabalho foi desenvolvido um ambiente para análise citomorfométrica em imagens extraídas de microscópios por meio da reconstrução 3D, permitindo a exploração das informações em um navegador web. O ambiente disponibiliza algoritmos

de pré-processamento das imagens, segmentação, renderização volumétrica, reconstrução de isosurferfícies, extração de atributos quantitativos baseando-se na citomorfometria, análise estatística e métodos de aprendizado de máquina. No desenvolvimento do ambiente proposto foi implementada uma arquitetura baseada no modelo cliente/servidor possibilitando a disponibilização na nuvem.

A citomorfometria ou alterações morfológicas nas estruturas celulares são importantes no auxílio dos métodos de diagnósticos. Ela pode fornecer informações relevantes para auxiliar os biólogos na predição dos comportamentos celulares. A citomorfometria também pode ser utilizada como uma metodologia para identificar a morte celular programada (apoptose) (ELMORE, 2007).

Foram realizados dois estudos de casos utilizando o ambiente proposto: o primeiro faz o uso da citomorfometria para classificação da localização de padrões subcelulares em células HeLa e o segundo faz o uso da citomorfometria para a análise das alterações no padrão de heterocromatina nos cérebros de tartarugas da Amazônia. Em ambos estudos de casos, o uso da citomorfometria possibilitou a extração de informações relevantes para o reconhecimento de padrões celulares e permitiu a validação do ambiente proposto.

## 1.1 Motivação

A maior parte dos softwares de bioimagens são executados como sistemas locais e necessitam de compatibilidade, instalação e configuração no sistema operacional do usuário. Alguns softwares disponibilizados na web dependem da instalação de algum *plug-in* ou não possuem recursos necessários para o processamento, visualização e análise quantitativa das amostras extraídas de microscópios.

As ferramentas baseadas na web facilitam a mobilidade, ou seja permitem o acesso em qualquer dispositivo e em qualquer lugar por meio da internet. A visualização de objetos 3D em um navegador web é dependente das tecnologias suportadas pelos navegadores. Algumas tecnologias não são suportadas ou necessitam de um *plug-in* pelo navegador web, tais como Java Applet, Adobe Flash e Adobe Flex. A instalação e compatibilidade de *plug-ins* pode ser um fator limitante para o uso de um software (JOURDAIN; AYACHIT; GEVECI, 2010; BEHR et al., 2009).

Um ambiente web para análise citomorfométrica pode auxiliar especialistas em diferentes áreas na avaliação de amostras e contribuir com a usabilidade dessas ferramentas. Com a análise citomorfométrica, por exemplo, poderá auxiliar especialistas na predição dos comportamentos celulares para auxiliar no diagnóstico precoce de doenças e também auxiliar biólogos na análise de alterações morfológicas das células durante o processo de morte celular.

## 1.2 Objetivos

Os objetivos desta dissertação foram divididos em geral e específicos.

### 1.2.1 Geral

Desenvolver um software de bioimagem na nuvem com capacidade de realizar operações de visualização computacional e análise citomorfométrica.

### 1.2.2 Específicos

- Desenvolver um ambiente de visualização utilizando Biblioteca de Gráficos Web (*Web Graphics Library*) (WebGL) para visualizar imagens 3D utilizando o modelo cliente/servidor;
- Prover ao ambiente técnicas de processamento digital de imagens e visualização computacional, além de métodos estatísticos e algoritmos de aprendizado de máquina;
- Fornecer algoritmos para análise citomorfométrica em objetos 3D para quantificação de parâmetros nucleares;
- Realizar estudos de casos para a validação do ambiente.

## 1.3 Contribuições

Com o desenvolvimento deste trabalho obteve-se as seguintes contribuições:

- Ambiente web de visualização 3D, de fácil utilização e acesso, uma vez que não é necessária a instalação do software ou de *plug-ins*, pois ele é acessível pela internet, permitindo que usuários, tais como, biólogos possam realizar os seus experimentos na nuvem;
- Levantamento bibliográfico de softwares de bioimagens livres e comparação entre suas características;
- Análise citomorfométrica em imagens microscópicas;
- Estudo de caso da utilização de citomorfometria para classificação da localização de padrões subcelulares em células HeLa;
- Estudo de caso da utilização da citomorfometria para a análise das alterações no padrão de heterocromatina nos cérebros de tartarugas da Amazônia.

- Publicação do artigo “Ambiente web para citomorfometria usando imagens 3D” no *Journal of Health Informatics*, Suplemento I do XV Congresso Brasileiro de Informática em Saúde - CBIS 2016.

## 1.4 Organização da Dissertação

O Capítulo 2 apresenta a fundamentação teórica, abordando conceitos sobre a visualização científica, análise morfométrica e aprendizado de máquinas.

O Capítulo 3 apresenta um estudo dos softwares de bioimagens livres existentes na literatura. Neste estudo foi realizado a comparação das características do BioWebVis com as propostas existentes na literatura.

O Capítulo 4 aborda o ambiente proposto (denominado BioWebVis) para a análise citomorfométrica utilizando imagens tridimensionais.

O Capítulo 5 apresenta um estudo de caso da análise citomorfométrica para localização de padrões subcelulares em isosuperfícies 3D em que o ambiente BioWebVis é utilizado em todo o *pipeline*.

O Capítulo 6 apresenta um estudo de caso da análise citomorfométrica das alterações no padrão de heterocromatina nos cérebros de tartarugas da Amazônia em que o ambiente BioWebVis é utilizado em todo o *pipeline*.

No Capítulo 7 é apresentada a conclusão desta pesquisa, bem como uma descrição das limitações encontradas e algumas perspectivas futuras.

# CAPÍTULO 2

---

## Fundamentação Teórica

Neste capítulo são abordados os conceitos sobre a visualização científica, análise morfométrica, reconhecimento de padrões e diversos trabalhos relacionados com esta dissertação.

### 2.1 Visualização de Informação e Científica

A Visualização de Informação (VI) é uma área da pesquisa com o foco na representação visual de dados. Para construir um modelo de VI, coleta-se e armazena os dados brutos, em seguida é feita a transformação dos dados em algo comprehensível e por fim a aplicação de um modelo para representação visual dos dados para a cognição do usuário (WARE, 2012).

A Visualização Científica (ViSC) faz o uso de técnicas para permitir a transformação de dados brutos em uma abstração geométrica que pode ser facilmente comprehensível para a exploração científica (HABER; MCNABB, 1990).

Na ViSC, o processo de análise e visualização é composto por um ciclo interativo de três atividades, a modelagem, a análise e a visualização, conforme apresenta a Figura 1.

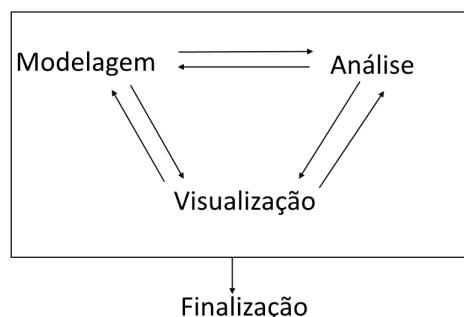


Figura 1 – Processo de análise e visualização. Figura adaptada de (GALLAGHER; PRESS, 1994).

A modelagem constrói um modelo que descreve o problema. Uma vez criado o modelo, análises podem ser inferidas e os resultados podem ser visualizados. O ciclo é realizado até que seja produzido um resultado satisfatório (GALLAGHER; PRESS, 1994).

Para Gallagher e Press (1994), a ViSC por meio de modelos 3D possibilita a visualização de dados em grande escala. Com a utilização de métodos para representação volumétrica e superfícies é possível analisar fenômenos complexos que as vezes não podem ser vistos em um modelo 2D. No processo de visualização tridimensional (3D) algumas técnicas podem ser utilizadas, como:

- ❑ **Renderização de resultados:** são técnicas como volumes, contornos e superfícies para criar representações de imagens;
- ❑ **Transformação de modelos:** capacidade de modificação da visão de um modelo, apresentando atributos dado uma perspectiva, distância e posição do observador;
- ❑ **Animação:** envolve movimentos e variações sobre o tempo. São exigidas técnicas para redesenhar as informações visuais com desempenho satisfatório;
- ❑ **Manipulação dos resultados:** são técnicas para apresentar resultados quantitativos de forma interativa para serem explorados.

Para Jung et al. (2012) um ambiente que fornece métodos para ViSC deve ser capaz de renderizar representações por meio dos algoritmos de visualização volumétrica, além de possibilitar de forma interativa a exploração dos dados.

Os algoritmos de visualização volumétrica são capazes de criar uma projeção de um conjunto de dados 3D em um plano de imagens 2D, para compreensão e exploração dos dados volumétricos. Eles estão divididos em duas categorias (ELVINS, 1992):

- ❑ **Renderização Volumétrica Direta (*Direct Volume Renderer – DVR*):** Cria imagens de um conjunto de dados que não apresentam uma forma geométrica definida. Assim, ele realiza um mapeamento dos elementos diretamente no plano sem utilizar primitivas geométricas para mediar a representação.
- ❑ **Isosuperfície:** Também conhecido como Ajuste de Superfície (*Surface-Fitting – SF*), extrai uma superfície em um volume por meio do ajuste de primitivas geométricas, como polígonos, para contornos de superfícies de valor constante.

O conjunto de dados volumétricos são manipulados como uma matriz de volume de voxels ou uma matriz de células. Um voxel é a representação de um valor em um espaço 3D, seria equivalente a um pixel em um espaço 2D (ELVINS, 1992).

### 2.1.1 Renderização Volumétrica Direta

O *Volume Ray-casting* é um método de DVR. Nesse algoritmo é realizado uma projeção de raios, em que eles iniciam-se da tela do observador e percorre cada pixel  $(i, j)$  no plano de imagens 2D que receberá a projeção do volume, conforme apresenta a Figura 2 (ELVINS, 1992).

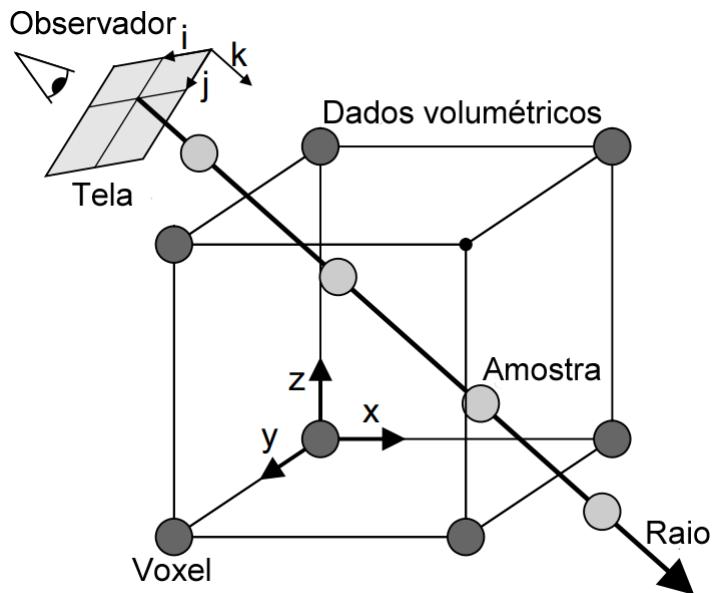


Figura 2 – Algoritmo *Volume Ray-casting*. Figura adaptada de (BOSMA; SMIT; LO-BREGT, 1998).

Quando um raio intercepta uma célula entre dois pontos de uma grade, interpolações podem ser realizadas para encontrar o ponto de intersecção, delimitando uma amostragem. Então uma função de transferência é invocada para encontrar a cor e opacidade. Assim, a cor é um gradiente sombreado e a opacidade é atenuada pela iluminação normalizada do gradiente. Por meio da iluminação do gradiente é possível ter uma indicação de um volume dentro de uma célula, então uma cor é atribuída ao pixel quando o raio encontra uma substância de volume. Ou seja, ao longo da projeção os valores de opacidades e cores são encontrados para fornecer o valor da opacidade e da cor de cada pixel (ELVINS, 1992).

### 2.1.2 Isosuperfície

O *Marching Cubes* (MC) é um método de extração de isosuperfície amplamente utilizado em imagens médicas. Esse algoritmo realiza triangulações em superfícies de densidade constante para extração de isosuperfícies em conjuntos de dados volumétricos (LORENSEN; CLINE, 1987).

O algoritmo MC utiliza o método de divisão e conquista para localizar a superfície em um cubo 3D criado a partir de duas fatias adjacentes ( $k$  e  $k + 1$ ), conforme apresenta a Figura 3.

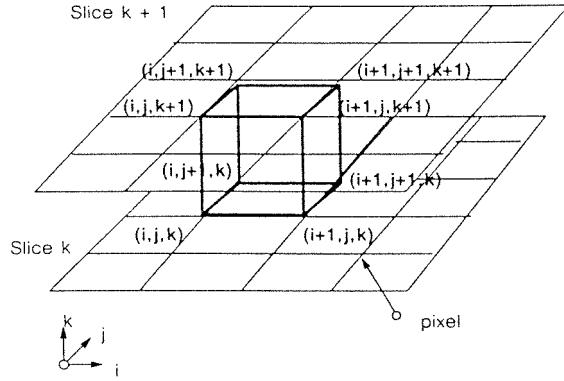


Figura 3 – *Marching Cubes*. Figura obtida de (LORENSEN; CLINE, 1987).

O algoritmo encontra a interseção da superfície com o cubo e atribui valor 1 para o vértice se o valor dos dados desse vértice é maior ou igual ao valor da superfície que está sendo construída. Valores abaixo do valor da superfície recebem zero e são descartados. Assim, foram definidos quatorze padrões de triangulações para reduzir a complexidade do problema, pois considerando que um cubo possui oito lados e temos cubos fora e dentro das superfícies, então teria uma possibilidade de  $2^8 = 256$  triangulações (LORENSEN; CLINE, 1987). A Figura 4 apresenta as quatorze triangulações. A triangulação 0 ocorre quando os valores estão acima ou abaixo do valor selecionado e não produz triangulação.

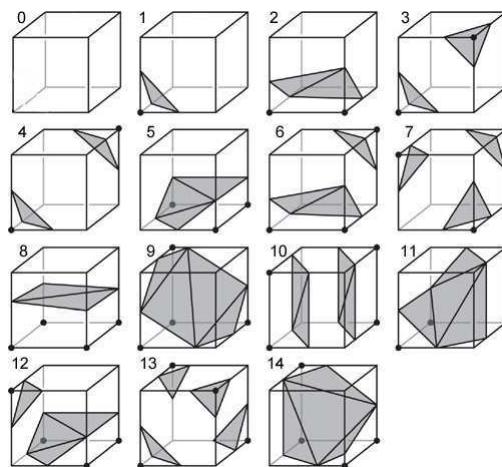


Figura 4 – Triangulações do *Marching Cubes*. Figura obtida de (LORENSEN; CLINE, 1987).

Ao fim, o algoritmo usa os gradientes de cada vértice de borda para encontrar as intersecções das bordas de superfície por meio de interpolação linear. Os gradientes

interpolados são utilizados para criar uma malha de triangulações que representam uma isosuperfície (ELVINS, 1992).

Na Figura 5(a) é possível visualizar um conjunto de imagens bidimensionais fatiadas, representando um conjunto de dados volumétricos. Após a aplicação do algoritmo *Volume Ray-casting* no conjunto de dados é possível visualizar representações volumétricas em que o volume é definido por voxels com tonalidades verdes, conforme mostra Figura 5(b). A representação de isosuperfície com *Marching Cubes* pode ser visualizado na Figura 5(c) (ELVINS, 1992).

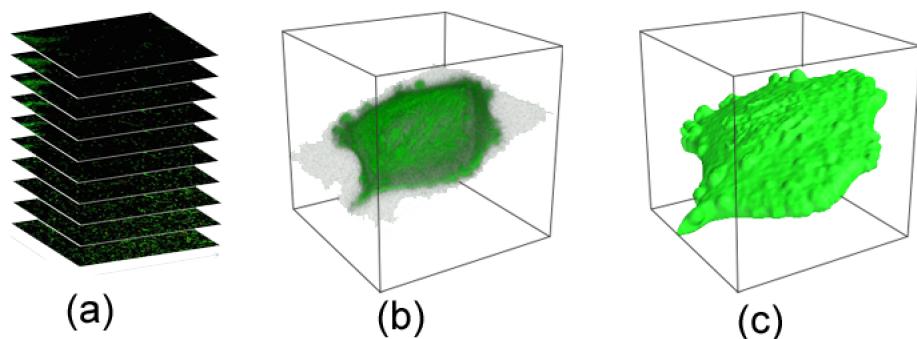


Figura 5 – Reconstrução 3D com o software BioImageXD. (a) Imagens fatiadas. (b) Visualização de volume. (c) Isosuperfície.

O método de *Marching Cubes* possui um desempenho melhor em relação ao método de *Volume Ray-casting*, pois ele não percorre todo o conjunto de dados a cada vez que a imagem é renderizada. Porém, o método *Marching Cubes* pode apresentar isosuperfícies não tão representativas de acordo com o modelo real, devido à reconstrução com falsos triângulos negativos e positivos na isosuperfície (ELVINS, 1992). O SF também não é apropriada quando os objetos a serem reconstruídos não possuem forma geométrica bem definida, como, por exemplo, no mapeamento de valores quantitativos dentro do volume, sendo, neste caso, mais apropriado o uso de DVR.

## 2.2 Análise Citomorfométrica

A morfometria é um conjunto de técnicas que permite descrever quantitativamente características estruturais dos objetos por meio da extração das informações, como a área da superfície, volume, circularidade e curvatura (WEIBEL, 1969). Na patologia, alterações morfológicas nas estruturas celulares são importantes no auxílio dos métodos de diagnósticos. A morfometria citológica ou citomorfometria é um conjunto de técnicas de morfometria para descrever características dos componentes celulares (WALDEMARIN; BELETTI; COSTA, 2004).

Em várias doenças, os componentes celulares como o nucléolo e a heterocromatina sofrem alterações estruturais que podem ser quantificadas. A caracterização das alterações pode fornecer informações úteis para o prognóstico e também predizer o andamento da doença (TRUE, 1996). Nas próximas subseções serão descritas algumas medidas que são utilizadas na citomorfometria. Todas são explicadas sob o ponto de vista computacional de como o cálculo da medida é realizado.

### 2.2.1 Área da Superfície

As técnicas de extração de isosuperfícies são capazes de extrair uma superfície por meio do ajuste de primitivas geométricas, como polígonos. A superfície extraída é composta por uma malha de triangulações (LORENSEN; CLINE, 1987).

A área da superfície é uma medida para estimar a área total da superfície de um objeto por meio do somatório de todas as áreas dos triângulos que compõe a malha do objeto (COSTA; JUNIOR, 2010; SCHROEDER; LORENSEN; MARTIN, 2004).

A área de um triângulo pode ser calculada utilizando as medidas de base e de altura. Quando não possui essas medidas, a área pode ser expressa por meio do tamanhos dos lados ( $a, b, c$ ) do triângulo utilizando a fórmula de Heron. A fórmula de Heron é expressa pela Eq. 1 (VAKIL, 1996).

$$K = \sqrt{s(s - a)(s - b)(s - c)}, \quad \text{onde } s = \frac{a + b + c}{2}. \quad (1)$$

A área de uma superfície pode ser calculada pelo somatório de todas as áreas dos triângulos que compõe a superfície do objeto, conforme apresenta a Eq. 2:

$$A = \sum_{i=0}^n K_i. \quad (2)$$

A variável  $n$  representa a quantidade de triângulos que compõe a superfície.

### 2.2.2 Volume

Para mensurar o volume de um objeto, o uso do Algoritmo do Teorema da Divergência (*Divergence theorem algorithm – DTA*) demonstrou superioridade em relação a contagem de voxels. Essa técnica estima o volume do objeto por meio do seu conjunto de pontos. A Eq. 3 é utilizada para estimar o volume com o DTA (ALYASSIN et al., 1994).

$$V = k_x \sum_i (x_i n_{x_i} \Delta a_i) + k_y \sum_i (y_i n_{y_i} \Delta a_i) + k_z \sum_i (z_i n_{z_i} \Delta a_i). \quad (3)$$

As variáveis  $x$ ,  $y$ , e  $z$  são as coordenadas,  $k_x$ ,  $k_y$ , e  $k_z$  são coeficientes estimados por meio do número de voxels com componentes normais nas direções  $x$ ,  $y$ , e  $z$  expressa como fração do número total de normais,  $n_{x_i}$  é o componente do vetor unitário normal na direção  $x$  e  $\Delta a_i$  é a máxima unidade do componente normal. A máxima unidade do

componente normal é obtida pela fração das dimensões dos voxels sobre os componentes normais do vetor unitário normal. Os coeficientes  $k_x$ ,  $k_y$ , e  $k_z$  são fatores de pesos do DTA para o cálculo do volume, a soma desses três atributos é sempre igual a 1, eles também são utilizados como atributos para a caracterização de um objeto (ALYASSIN et al., 1994; XIAOJUN et al., 2010).

### 2.2.3 Convexidade

O estimador de convexidade analisa irregularidades nos objetos. Um objeto é considerado suave (forma com nenhuma ou pouca irregularidade) quando a convexidade está mais perto de 1 ou irregular quando está mais perto de 0. A convexidade é estimada pela razão entre a área da superfície do objeto convexo e a área da superfície do objeto em análise. Pode-se também utilizar, ao invés da área, a medida de volume para estimar a convexidade. A convexidade pode ser calculada pelas fórmulas apresentadas na Eq. 4 (ZUNIC; ROSIN, 2004; KASSIMI et al., 2011).

$$C_a = \frac{A_{ch}}{A} \quad e \quad C_v = \frac{V_{ch}}{V}. \quad (4)$$

$C_a/C_v$  são coeficientes de convexidade baseada na área/volume da superfície,  $A_{ch}/V_{ch}$  é a área/volume da superfície do objeto convexo e  $A/V$  é a área/volume da superfície do objeto em análise (KASSIMI et al., 2011).

Para estimar  $A_{ch}$  e  $V_{ch}$ , inicialmente deve ser gerado o *convex hull* a partir do conjunto de pontos do objeto em análise. Um objeto  $S$  é convexo se os pontos  $A$  e  $B$  pertencem a  $S$  e se todos os pontos de um segmento de linha  $[AB]$  também pertencem a  $S$ . O menor conjunto convexo que inclui o objeto  $S$  é denominado *convex hull* do objeto e é expresso por  $CH(P)$ , conforme ilustrado na Figura 6 (ZUNIC; ROSIN, 2004).

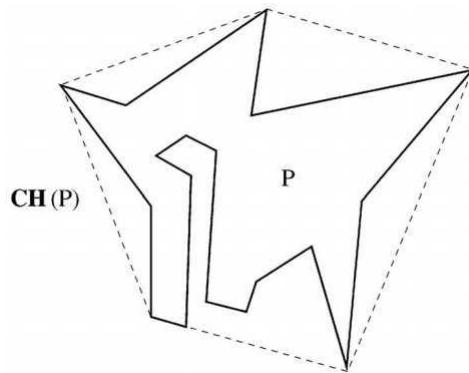


Figura 6 – Representação de um polígono  $P$  não convexo e seu *convex hull*  $CH(P)$ . Figura obtida de (ZUNIC; ROSIN, 2004).

O objeto 3D obtido pelo algoritmo MC (e.g. Figura 7(a)) é composto por uma malha de triângulos, para obter o *convex hull* desse objeto pode ser utilizado o método de trian-

gulação de Delaunay. Esse método é capaz de calcular triangulações em um conjunto de pontos no espaço 3D para um *convex hull* (e.g. Figura 7(b)). A triangulação de Delaunay maximiza o ângulo mínimo sobre todas as triangulações de um conjunto de pontos  $P$ , deixando os triângulos mais equiláteros (FISHER, 2004; SCHROEDER; LORENSEN; MARTIN, 2004).

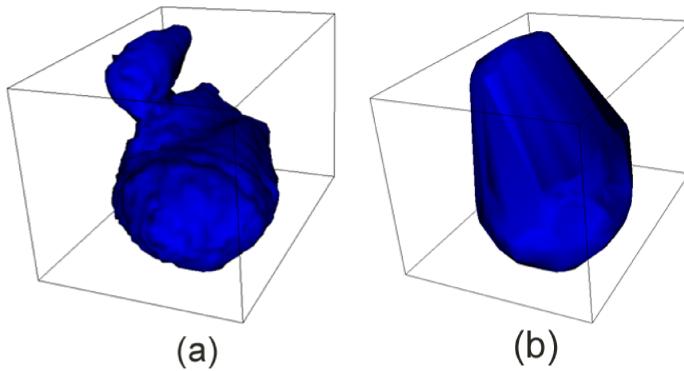


Figura 7 – *Convex hull* de uma célula. (a) Superfície de uma célula. (b) Superfície de uma célula após a triangulação de Delaunay.

Após a extração do *convex hull* do objeto, os coeficientes de área e volume da superfície podem ser extraídos e a convexidade mensurada.

## 2.2.4 Esfericidade

A esfericidade é uma medida que caracteriza o desvio da forma de um objeto a partir de uma esfera. Ela pode ser obtida por meio da área da superfície e do volume (SCHROEDER; LORENSEN; MARTIN, 2004). O Índice de Forma Normalizado (*Normalized Shape Index* – NSI) é uma medida de esfericidade disponível no VTK (SCHROEDER; LORENSEN; MARTIN, 2004). O NSI é sempre maior ou igual a 1 – se for igual a 1, o objeto é uma esfera. O NSI pode ser calculado pela Eq. 5:

$$NSI = (\sqrt{A}/V^{\frac{1}{3}})/2.199085233. \quad (5)$$

No método descrito por Kassimi et al. (2011) para estimar a esfericidade utiliza a Eq. 6.

$$S = 4.84 * V^{2/3}/A. \quad (6)$$

O  $A$  e  $V$  representam respectivamente a área da superfície e o volume do objeto.

### 2.2.5 Diâmetro Esférico Equivalente

O diâmetro esférico equivalente é uma medida utilizada para definir o tamanho de um objeto 3D. Ele pode ser obtido pela Eq. 7 (KASSIMI et al., 2011).

$$DEE = \left( \frac{4}{3} \right) * \pi * \sqrt{\left( \frac{A_{ch}}{\pi} \right)^3} \quad (7)$$

A variável  $A_{ch}$  é obtida pela a área da superfície do objeto convexo.

### 2.2.6 Curvatura

A curvatura é uma medida extraída por meio do contorno de uma superfície e expressa bem a natureza geométrica local de uma curva, pois ela é invariante de rotações, translações e reflexões (COSTA; JUNIOR, 2010). A extração dessa medida auxilia a compreensão da topologia das estruturas de uma superfície de modo que a classificação de um objeto pode ser inferida pela sua curvatura. Em uma superfície composta por uma malha de triangulações, o estimador de curvatura Gaussiana discreta tem sido uma ferramenta importante para a análise de curvaturas (MESMOUDI; FLORIANI; MAGILLO, 2012).

Para estimar a curvatura Gaussiana discreta de uma superfície, o teorema de Gauss-Bonnet é utilizado para produzir aproximações de curvaturas ao longo da vizinhança de um vértice. Assim, a curvatura é considerada constante em uma pequena vizinhança de um vértice. O teorema de Gauss-Bonnet infere a curvatura pela diferença entre os desvios dos ângulos e a área da vizinhança de um ponto  $p$ . Considere, por exemplo, a Figura 8 uma superfície de um objeto 3D em que os pontos  $p$ ,  $a_i$ ,  $a_{i+1}$  e  $a_{i-1}$  não estão no mesmo plano (MESMOUDI; FLORIANI; MAGILLO, 2012; MEYER et al., 2003).

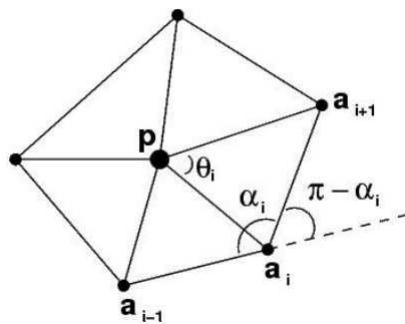


Figura 8 – Representação da superfície para inferir a curvatura Gaussiana no ponto  $p$ .  
Figura obtida de (MESMOUDI; FLORIANI; MAGILLO, 2012).

Para estimar a curvatura do ponto  $p$  é calculado o défice do ângulo, que consiste no somatório dos ângulos dos vértices adjacentes ao ponto  $p$  em fração da área da vizinhança.

A curvatura de um vértice é expressa pela Eq. 8 (MESMOUDI; FLORIANI; MAGILLO, 2012).

$$K_g = \frac{2\pi - \sum_{i=1}^F \theta_i}{A}. \quad (8)$$

Assim,  $K_g$  é a curvatura Gaussiana,  $A$  é a área da vizinhança (somatório da área de todos os triângulos incidentes) de um ponto  $p$ ,  $\theta_i$  é o ângulo em  $p$  formado pelo  $i$ -ésimo triângulo incidente em  $p$  e  $F$  é a quantidade de faces adjacentes ao ponto  $p$ . Para melhorar o desempenho, o estimador retorna zero para qualquer superfície plana e para qualquer vizinhança da superfície em formato de um anel (MEYER et al., 2003). Os coeficientes extraídos das curvaturas de um objeto podem fornecer uma série de medidas estatísticas, como média da curvatura, entropia, variância, entre outras (COSTA; JUNIOR, 2010). A Tabela 1 apresenta as medidas que podem ser extraídas a partir dos coeficientes de curvatura.

Tabela 1 – Medidas estatísticas sobre os coeficientes de curvatura. Nas fórmulas,  $N$  é a quantidade de coeficientes de curvatura extraídos,  $X_i$  é cada coeficiente de curvatura,  $\bar{X}$  a média dos coeficientes e  $P(X_i)$  a probabilidade do evento  $X_i$  ocorrer.

Nome	Fórmula
Média	$\frac{1}{N} \sum_{i=1}^N X_i$
Mediana	$X_k$ , onde: $k = \begin{cases} (N+1)/2 & \text{se } N = \text{ímpar} \\ (N/2 + (N/2 + 1))/2 & \text{caso contrário} \end{cases}$
Desvio Padrão	$\sqrt{\frac{\sum_{i=1}^N (X_i - \bar{X})^2}{N-1}}$
Entropia	$-\sum_{i=1}^N P(X_i) \log P(X_i)$

Na fórmula da mediana, se  $N$  for ímpar resultará na ordem do termo central dos coeficientes de curvatura, se par resultará na média dos dois termos centrais.

## 2.3 Reconhecimento de Padrões

O reconhecimento de padrões é uma área de pesquisa utilizada para a descoberta automática de padrões nos dados utilizando métodos computacionais. Os padrões descobertos são utilizados para classificar amostras em diferentes categorias. Assim, vários algoritmos de Aprendizado de Máquina (*Machine Learning* – ML) foram propostos para a resolução de problemas de reconhecimento de padrões. Esses algoritmos utilizam uma base de treinamento para aprender os seus padrões e classificar as novas amostras. Uma boa prática

é realizar um pré-processamento dos dados, ou seja, utilizar os atributos extraídos a fim de transformá-los em um novo espaço de variáveis, reduzindo a complexidade do modelo em análise (BISHOP, 2007).

### 2.3.1 Normalização

Os atributos morfométricos podem ser extraídos em diferentes escalas de mensuração. No trabalho proposto por Bacus et al. (1999), foi utilizado o normalizador *Z-Score* para criar uma escala comum de classificação nuclear para os diferentes dados quantitativos extraídos dos componentes celulares. O *Z-Score* utiliza a média ( $\mu$ ) e o desvio padrão ( $\sigma$ ) dos dados brutos para indicar o número de desvios que a amostra está acima ou abaixo da média. O  $Z\text{-Score}(z)$  de um atributo  $X$  pode ser calculado pela fórmula apresentada na Eq. 9.

$$z = \frac{X - \mu}{\sigma}. \quad (9)$$

Após a normalização, os dados estarão na mesma escala, facilitando o reconhecimento de padrões no algoritmo de ML.

### 2.3.2 Seleção de Atributos

A seleção de atributos é uma fase de pré-processamento dos dados para a redução da dimensionalidade de um modelo. Um modelo com muitos atributos possui uma complexidade computacional maior e a seleção de atributos permite também projetar um modelo de características que pode melhorar o desempenho de um algoritmo de ML.

Nessa etapa deve ser selecionados atributos que melhor discriminam as classes do modelo, atributos que possuem uma grande variação entre as classes e uma baixa variação dentro da classe, e atributos com baixa correlação entre os outros atributos do modelo.

Assim, deve-se analisar cada atributo e descartar os atributos com pouca capacidade discriminatória. Para isso, pode-se utilizar métodos estatísticos, como análise de variância, análise de regressão, correlação de Pearson, métodos baseados em árvore de decisões, como *Extra tree* e Análise de Componentes Principais (*Principal Component Analysis – PCA*) (THEODORIDIS; KOUTROUMBAS, 2006; BISHOP, 2006).

O *Extra tree* é um algoritmo de classificação baseado em árvores de decisão. Em cada nó da árvore é selecionado um subconjunto de atributos aleatoriamente para a regra de decisão. O usuário deve inicialmente informar o número de árvores ( $n$ ) para a construção da floresta. Para a construção de cada árvore  $T$  é selecionado todo o conjunto de dados com  $m$  atributos selecionados aleatoriamente. Do subconjunto  $m$  de atributos, um atributo é selecionado aleatoriamente como variável de divisão. Em cada nível da árvore o conjunto de dados de entrada é dividido em dois (árvore binária). Em seguida, para

cada nó da árvore, a seleção do conjunto de dados com  $m$  atributos, a seleção do atributo aleatório, e a divisão dos dados são realizadas recursivamente até a finalização da árvore.

Para cada atributo selecionado como variável de divisão é calculado o fator de importância. Com base no fator de importância os atributos mais relevantes são selecionados, para isso, é selecionado um limiar com base na média dos fatores de importância dos atributos. Se o fator de importância do atributo é maior que o limiar, então o atributo é selecionado, senão, ele é descartado (GEURTS; ERNST; WEHENKEL, 2006; SUBRAMANIAN, 2015).

O PCA identifica novas variáveis, denominadas de componentes principais, oriundas de combinações lineares das variáveis originais. O objetivo é criar um conjunto novo de variáveis linearmente não correlacionadas (RINGNÉR, 2008). O PCA realiza uma projeção ortogonal dos dados em um espaço linear dimensional menor, denominado como subespaço principal (denotada pela linha magenta na Figura 9), de modo que a projeção ortogonal dos dados (denotado pelos pontos vermelhos na Figura 9) nesse subespaço maximize a variância da projeção dos dados (denotado pelos pontos verdes na Figura 9). A linha azul na Figura 9 denota a minimização da soma dos quadrados dos erros da projeção (BISHOP, 2006).

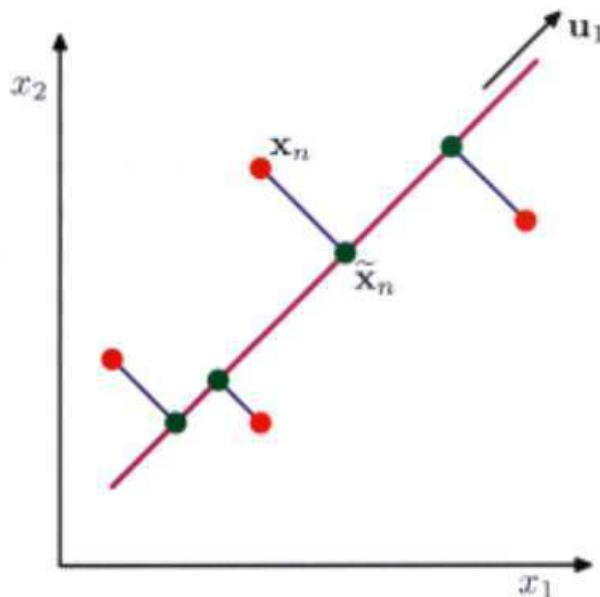


Figura 9 – PCA. Figura obtida de (BISHOP, 2006).

### 2.3.3 Aprendizado de Máquina

Os algoritmos de ML utilizam de métodos computacionais para classificar objetos em um número de categorias ou classes. Esses algoritmos utilizam uma base de treinamento para descobrir padrões em um conjunto de amostras. É realizado uma representação do

conhecimento prévio, em que as características são utilizadas para mapear as amostras em um espaço vetorial. As classes similares formam grupos localizados nesse espaço. A capacidade de classificar corretamente novas amostras que diferem das utilizadas para treinamento é conhecimento como generalização. O objetivo de um algoritmo de ML é maximizar a generalização, ou seja, distinguir os grupos localizados no espaço vetorial para realizar previsões precisas das novas amostras (THEODORIDIS; KOUTROUMBAS, 2006; BISHOP, 2006). A Figura 10 apresenta a representação e a generalização no aprendizado de máquina.

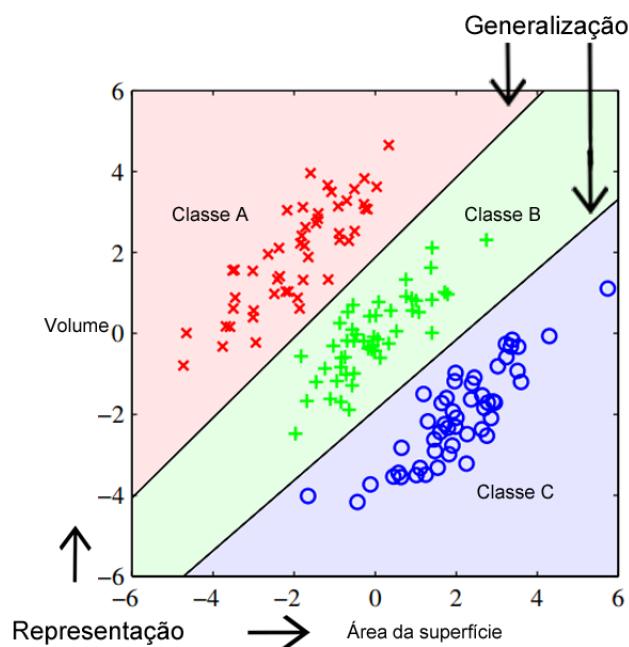


Figura 10 – Aprendizado de máquina. Figura adaptada de (BISHOP, 2006).

Os algoritmos de aprendizado de máquina estão divididos em dois grupos (THEODORIDIS; KOUTROUMBAS, 2006):

- **Aprendizado supervisionado:** são algoritmos que possuem o conhecimento prévio das categorias das amostras da base de treinamento .
- **Aprendizado não supervisionado:** são algoritmos que não possuem o conhecimento prévio das categorias das amostras da base de dados. Seu objetivo é descobrir grupos similares (também denominados de *cluster*) dentro da base de dados.

Para que os classificadores supervisionados possam aprender o modelo de dados e classificar corretamente novas amostras, diversas estratégias podem ser utilizadas. Elas estão organizadas nas seguintes categorias (THEODORIDIS; KOUTROUMBAS, 2006):

- **Classificadores baseados na teoria de decisão de Bayes:** utiliza a probabilidade condicional do teorema de Bayes para criar um modelo de distribuição para reconhecimento estatísticos dos padrões a serem explorados.
- **Classificadores de lineares:** a classificação é realizada por superfícies de decisões lineares (hiperplano), ou seja, funções lineares são definidas para criar um separador linear no espaço vetorial.
- **Classificadores baseados em instâncias:** realiza o aprendizado utilizando as características das instâncias (amostras), tal como, localização no plano vetorial.
- **Redes neurais:** utiliza conceitos da biologia para formar um conjunto de unidades de processamento (neurônios) com conexões entre eles (sinapse), em que são atribuídos pesos nas conexões entre neurônios. Cada neurônio possui uma regra de ativação, criando um hiperplano de decisão.

O ambiente proposto nesta dissertação disponibiliza algoritmos de classificação supervisionada em cada uma das categorias supracitadas, pois, um determinado algoritmo de classificação poderá ter um desempenho diferente para cada problema. Assim, são disponibilizados os seguintes classificadores: o Naive Bayes (detalhes podem ser encontrados em (RISH, 2001)) e o QDA (detalhes podem ser encontrados em (SRIVASTAVA; GUPTA; FRIGYIK, 2007)) como classificadores baseados na teoria de decisão de Bayes; o Máquina de Vetor Suporte (*Support Vector Machine*) (SVM) (detalhes podem ser encontrados em (HEARST et al., 1998)) como classificador não probabilístico linear; o *K-Nearest Neighbors* (K-NN) (detalhes podem ser encontrados em (CUNNINGHAM; DELANY, 2007)) como classificador baseado em instâncias que utiliza a distância entre as amostras; e o *Multilayer Perceptron* (detalhes podem ser encontrados em (THEODORIDIS; KOUTROUMBAS, 2006)) como classificador baseado em Redes Neurais.

O método de validação cruzada *k-fold* pode ser utilizado para avaliar o desempenho de um classificador supervisionado. Ele divide a base de dados em  $k$  subconjuntos mutualmente exclusivos do mesmo tamanho. Um subconjunto é utilizado para teste, e os  $k - 1$  restantes são utilizados para o treinamento. Esse processo é repetido  $k$  vezes, em cada teste o subconjunto é alterado, a fim de cobrir todo o subconjunto (PIROOZNIA et al., 2008). Ao final calcula-se a acurácia. A acurácia (*accuracy*) expressa o número de previsões corretas do classificador, ou seja, a relação entre o número de previsões corretas (*correct*) e o número total de previsões (*predictions*), conforme a Eq. 10 (JENI; COHN; TORRE, 2013).

$$\text{accuracy} = \frac{\text{correct}}{\text{predictions}} \quad (10)$$

A acurácia é amplamente utilizada para medir o desempenho de um classificador supervisionado. No entanto, se as probabilidades das classes são muito diferentes, ela pode

ser enganosa. Assim a melhor escolha é o *F1-score*. Para definir essa medida, é preciso rotular a classificação de cada objeto de acordo com as seguintes categorias: verdadeiros positivos (*truepositive*), que são objetos corretamente rotulados como positivos; falsos positivos (*falsepositive*), que são objetos negativos incorretamente rotulados como positivos; e falsos negativos(*falsenegative*), que são objetos positivos incorretamente rotulados como negativos. O *F1-score* representa a média ponderada entre a precisão (*precision*) e a sensibilidade (*recall*), conforme a Eq. 11 (JENI; COHN; TORRE, 2013).

$$F1 = 2 * \frac{precision * recall}{precision + recall} \quad (11)$$

A precisão mede o potencial do classificador não predizer como positivo os objetos negativos, Eq. 12.

$$precision = \frac{truepositive}{truepositive + falsepositive} \quad (12)$$

A sensibilidade indica o potencial do classificador em encontrar todas as amostras positivas, Eq. 13

$$recall = \frac{truepositive}{truepositive + falsenegative} \quad (13)$$

O ambiente proposto disponibiliza o algoritmo de classificação não supervisionada denominado *K-means*. O objetivo do *K-means* é particionar o conjunto de dados em um número  $k$  de grupos (*clusters*). O valor de  $k$  deverá ser informado como entrada do algoritmo. O algoritmo seleciona  $k$  centroides e para cada centroide ele encontra uma atribuição de pontos de dados de modo que a soma dos quadrados das distâncias de cada ponto de dados para o centroide seja mínima. As fases de reatribuição dos pontos de dados e a reconfiguração do centroide são repetidas até que não haja mais alterações nas atribuições ou até exceder o número máximo de iterações (THEODORIDIS; KOUTROUMBAS, 2006).

O *V-measure* é utilizado avaliar o desempenho de um algoritmo não supervisionado. Essa medida fornece uma solução para problemas encontrados em outras medidas para avaliar o desempenho de algoritmos não supervisionados, tais como, dependência do algoritmo de agrupamento, problema da correspondência (os agrupamentos de apenas uma parte dos pontos de dados são avaliados) e avaliação precisa com a combinação de dois aspectos desejáveis de agrupamento, homogeneidade e completude. O *V-measure* é calculado pela média ponderada entre a homogeneidade (*homogeneity*) e a completude (*completeness*), conforme a Eq. 14 (ROSENBERG; HIRSCHBERG, 2007).

$$V = 2 * \frac{homogeneity * completeness}{homogeneity + completeness} \quad (14)$$

Para satisfazer o critério de homogeneidade, um *cluster* ( $K$ ) deve consistir apenas em amostras de uma classe ( $C$ ), Eq. 15 (ROSENBERG; HIRSCHBERG, 2007).

$$\text{homogeneity} = \begin{cases} 1 & \text{se } H(C) = 0 \\ 1 - \frac{H(C|K)}{H(C)} & \text{caso contrário} \end{cases} \quad \text{em que :}$$

$$H(C|K) = - \sum_{k=1}^{|K|} \sum_{c=1}^{|C|} \frac{a_{ck}}{N} \log \frac{a_{ck}}{\sum_{c=1}^{|C|} a_{ck}} \quad (15)$$

$$H(C) = - \sum_{c=1}^{|C|} \frac{\sum_{k=1}^{|K|} a_{ck}}{N} \log \frac{\sum_{k=1}^{|K|} a_{ck}}{N}$$

A variável  $a_{ck}$  representa o número de elementos da classe  $c$  no *cluster*  $k$ . Para satisfazer o critério da completude, todas as amostras da mesma classe ( $C$ ) deverão ser concentradas em *clusters* ( $K$ ) únicos, Eq. 16 (ROSENBERG; HIRSCHBERG, 2007).

$$\text{completeness} = \begin{cases} 1 & \text{se } H(K) = 0 \\ 1 - \frac{H(K|C)}{H(K)} & \text{caso contrário} \end{cases} \quad \text{em que :}$$

$$H(K|C) = - \sum_{k=1}^{|K|} \sum_{c=1}^{|C|} \frac{a_{ck}}{N} \log \frac{a_{ck}}{\sum_{k=1}^{|K|} a_{ck}} \quad (16)$$

$$H(K) = - \sum_{k=1}^{|K|} \frac{\sum_{c=1}^{|C|} a_{ck}}{N} \log \frac{\sum_{c=1}^{|C|} a_{ck}}{N}$$

O *V-measure* varia de 0 a 1. Em que 1 indica o perfeito agrupamento.

# CAPÍTULO 3

## Trabalhos Relacionados

Com a evolução dos meios de aquisição de imagens microscópicas, diversas ferramentas para análise e visualização 3D foram desenvolvidas para auxiliar cientistas na exploração dos dados. Nesta seção serão apresentados softwares de bioimagens que são executados localmente e também pela web. Foram selecionados os softwares de bioimagens livres e gratuitos, não foram analisados os softwares pagos devido a dificuldade em obtê-los. Também será apresentado nesta seção algumas características do BioWebVis, para facilitar a comparação com as outras propostas existentes na literatura (a descrição completa do BioWebVis é apresentada no Capítulo 4).

### 3.1 Softwares de bioimagens de execução local

A maioria dos softwares de bioimagens são disponibilizados como aplicações locais, ou seja, são instalados e executados pelo sistema operacional do computador do usuário, tais como, BioimageXD, Vaa3D, ImageJ, Fiji, 3D Slicer, ICY e CellProfiler. Segundo (ELI-CEIRI et al., 2012), o BioimageXD e o Vaa3D oferecem as melhores ferramentas para visualização 3D.

O objetivo do BioImageXD é fornecer métodos para análise e visualização 3D. Ele possui cerca de 220 métodos para o processamento de imagens, 17 métodos para segmentação e módulos para criar renderizações de volume por meio de fatias 2D (e.g., Figura 11(a)), reconstrução de superfície, medições 3D, separação de objetos, entre outros. Por meio da reconstrução de isosuperfície com o algoritmo *marching cubes* é possível extrair informações quantitativas como distância e ângulo das superfícies (e.g., Figura 11(b)). O módulo de visualização permite ao usuário utilizar recursos de zoom, interação e seleção de área de interesse (KANKAANPÄÄ et al., 2012).

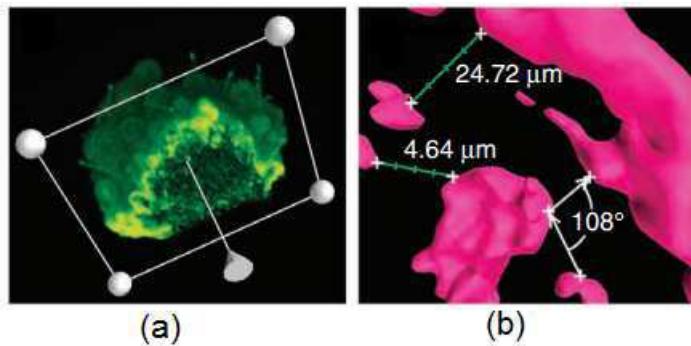


Figura 11 – BioImageXD. (a) Renderização de volume. (b) Reconstrução de superfície e cálculo de distâncias e ângulo. Figura obtida de (KANKAANPÄÄ et al., 2012).

O Vaa3D é uma aplicação de visualização e análise de imagens 3D voltado para neurobiologia (ELICEIRI et al., 2012). Ele foi projetado para trabalhar com dados em grande escala, permitindo a visualização de volume e isosuperfícies. Sua arquitetura é baseada em *plug-ins* permitindo integração com o ImageJ (PENG et al., 2014).

Para Eliceiri et al. (2012) o ImageJ é uma das principais ferramentas para a análise de imagens, devido a sua arquitetura baseada em *plug-ins* e pela comunidade que difundiu centenas de *plug-ins* em diferentes áreas de aplicação. Ele possui algumas limitações em relação análise multidimensional. Assim, foi criado o Fiji, uma versão aprimorada do ImageJ que permite análise de dados multidimensionais. O Fiji possui diversos *plug-ins*, métodos de visualização de volume e isosuperfície, e permitir a extração de informações quantitativas (COLLINS et al., 2007; SCHINDELIN et al., 2012).

O 3D Slicer fornece métodos de registro, segmentação, renderização de volumes e extração de isosuperfícies para a manipulação de imagens médicas (PIEPER; HALLE; KIKINIS, 2004). O ICY é uma plataforma colaborativa para compartilhar materiais e métodos de bioimagem. O ICY disponibiliza algoritmos de processamento, visualização e análise. Ele é baseado em *plug-ins* e permite a integração com o ImageJ (CHAUMONT et al., 2012). O CellProfiler é um software para a análise quantitativa de bioimagens, ele fornece diversos algoritmos para processamento das imagens, extração de características e algoritmos de aprendizado de máquina (LAMPRECHT et al., 2007).

Os softwares ImageJ, Fiji e ICY compartilham a mesma arquitetura de *plug-ins*, assim eles possuem diversos algoritmos desenvolvidos pela comunidade, criando uma ampla gama de funcionalidades. Ao longo dos anos diversas ferramentas de análise e visualização 3D foram desenvolvidas. Os cientistas necessitam de uma variedade de métodos computacionais para realizarem os seus experimentos. Uma ferramenta pode não fornecer todos esses métodos, assim as ferramentas devem permitir a integração com outras ferramentas, ou seja, a saída da ferramenta poderá ser a entrada em outra ferramenta (ELICEIRI et al., 2012).

Nos softwares de bioimagens disponibilizados como aplicações locais em alguns sistemas operacionais é necessário que o usuário tenha conhecimento em computação para a instalação e atualização da aplicação. Algumas aplicações não fornecem mecanismos para atualização automática e possuem poucos recursos para análise citomorfométrica, muitos algoritmos ficam dependentes da disponibilização e instalação de algum *plug-in*. Isso pode limitar o uso dessas ferramentas por alguns profissionais que não possuem conhecimento em computação e necessitam de algum algoritmo para a análise citomorfométrica.

## 3.2 Softwares de bioimagens na nuvem

Trabalhos como *OME Remote Objects* (OMERO), *Bio-Image Semantic Query User Environment* (Bisque), Slicer Drop, BioSig3D e *BioImage Mining, Analysis and eXploration* (BioIMAX) oferecem acesso por meio de um navegador web. O OMERO é uma plataforma para gerenciar experimentos biológicos. Sua principal característica é fornecer um meio de armazenamento na nuvem, suportando mais de 100 tipos de formatos de imagens microscópicas. Para facilitar a manipulação das imagens, os formatos são padronizados em OME-XML ou em OME-TIFF, eles podem ser convertidos para outros formatos com a biblioteca Bio-Formats. O OMERO é uma aplicação baseada na arquitetura cliente/servidor, as imagens são armazenadas em um servidor e qualquer cliente pode recuperar informações por meio de uma Interface de Programação de Aplicação (*Application Programming Interface – API*) (ALLAN et al., 2012).

O OMERO disponibiliza um cliente web para visualização das imagens no servidor, permitindo que o cientista visualize imagens 3D utilizando fatias ortogonais. O objetivo do OMERO não é fornecer métodos de análise e visualização, uma vez que possui outras ferramentas com esse propósito (ALLAN et al., 2012).

O Bisque é um ambiente web para gestão, compartilhamento, análise e extração de dados quantitativos em imagens multidimensionais. Ele fornece algoritmos para segmentação e detecção de centroides em imagens 3D. Para utilizar funcionalidades não implementadas pelo Bisque, ele possui compatibilidade com outros softwares, como, ImageJ, CellProfiler e BioView3D. No ambiente web ele permite a visualização 3D utilizando fatias ortogonais. Para fornecer uma visualização volumétrica mais detalhada, o objeto em análise deve ser exportado para uma aplicação externa denominada BioView3D, executada como um sistema local (KVILEKVAL et al., 2010).

O BioIMAX proposto por Loyek et al. (2011), é um ambiente web desenvolvido em Adobe Flex voltado para exploração e mineração de dados, necessitando do *plug-in* do Adobe Flash instalado no navegador. Ele permite visualizar imagens em planos 2D e gráficos de histograma.

O Slicer Drop é um software web com foco para as imagens médicas. Ele possui algoritmos de visualização 3D que podem ser aplicados em imagens biológicas. O seu foco

é a visualização de dados científicos em 2D e 3D, sem a necessidade de processamento em um servidor, utilizando a tecnologia WebGL (HAEHN et al., 2014).

O ParaViewWeb é um *framework* voltado para a visualização na web baseado na arquitetura cliente/servidor, não sendo considerado um software de bioimagem. Ele utiliza JavaScript para realizar as chamadas no servidor e então renderizar o objeto com WebGL. Sua arquitetura permite a renderização de uma grande quantidade de dados e demonstra-se uma arquitetura promissora (JOURDAIN; AYACHIT; GEVECI, 2010)

O BioSig3D é um ambiente web para visualização de modelos de culturas celulares. Ele fornece uma solução de armazenamento, segmentação, visualização 3D e análise quantitativa. O BioSig3D utiliza componentes do OMERO, tais como, Bio-Formats e OME Image Server, e utiliza o ParaViewWeb para a renderização 3D utilizando WebGL. Com a arquitetura Cliente/Servidor, o servidor realiza a criação de um arquivo de visualização 3D (.vti) e o envia para o ParaViewWeb renderizar. Esse ambiente disponibiliza a extração de mais de 50 atributos, categorizados em morfométricos, localização e textura (BILGIN et al., 2016).

Os softwares de bioimagens baseados na web, com exceção do BioSig3D, não fornecem métodos suficientes para realizar a análise citomorfométrica e visualização com imagens 3D. Um diferencial da proposta deste trabalho em relação aos trabalhos já existentes, é a possibilidade de realizar a análise citomorfométrica e visualização 3D em imagens microscópicas no navegador, sem o uso de *plug-ins*, melhorando a usabilidade e permitindo o uso em diferentes dispositivos, como *smartphones* e *tablets*. Para isso, algoritmos de pré-processamento, segmentação, visualização (*ray-casting* e *marching cubes*), análise citomorfométricas, métodos estatísticos e algoritmos de ML são disponibilizados.

### 3.3 Comparação com trabalhos relacionados

BioWebVis pode ser comparado com os softwares de bioimagens na nuvem, como, Bisque, Omero, BioIMAX, Slicer Drop e BioSig3D e com os softwares de bioimagens locais, como, BioimageXD, Vaa3D, ImageJ, Fiji, 3D Slicer, ICY e CellProfiler. Para a comparação com os trabalhos as características foram divididas em 4 categorias: Visualização multidimensional, processamento de imagem, análise quantitativa e arquitetura. As características foram selecionadas com base nas funcionalidades dos softwares de bioimagem descritas na literatura.

A Tabela 2 apresenta uma comparação das características de visualização multidimensional do trabalho proposto com os trabalhos relacionados citados anteriormente. A coluna “Tempo” na Tabela 2 indica se o ambiente permite a visualização da evolução das imagens ou dos volumes no tempo.

Tabela 2 – Comparação da visualização multidimensional dos trabalhos relacionados.

	Dimensões	Multicanais	Tempo	Renderização volumétrica	Extração de Isosuperfície
Trabalho proposto	3D	X		X	X
Bisque	3D	X	X	X	X
Omero	3D				
BioIMAX	3D				
Slicer Drop	3D			X	
BioSig3D	3D			X	X
BioimageXD	3D	X		X	X
Vaa3D	3D	X	X	X	X
ImageJ	3D	X	X	X	X
Fiji	3D	X	X	X	X
3D Slicer	3D	X	X	X	X
ICY	3D	X	X	X	X
CellProfiler	2D				

Os ambientes locais disponibilizam para a visualização multidimensional uma gama maior de funcionalidades em comparação com os ambientes na nuvem. A interação com elementos 3D em uma aplicação na nuvem é dependente das tecnologias de desenvolvimento web. Atualmente, a tecnologia WebGL para visualização 3D no navegador é mais aconselhável, pois essa tecnologia é nativa no navegador com suporte ao HTML 5, não necessitando de instalação de *plug-ins*. Porém, essa tecnologia é recente, lançada em 2011, estando presente em poucos softwares de bioimagem na nuvem.

As aplicações Bisque e SlicerDrop emitem erros ao realizar a renderização volumétrica de grandes conjuntos de dados, devido ao processamento dos algoritmos serem executados no lado do cliente (processamento ocorre no navegador utilizando JavaScript). O ambiente proposto não fornece um método de visualização da evolução de um volume sobre o tempo.

A Tabela 3 apresenta uma comparação dos algoritmos de processamento de imagens do trabalho proposto com os trabalhos relacionados.

Tabela 3 – Comparaçāo dos algoritmos de processamento dos trabalhos relacionados.

	Segmentação	Redução de ruído	Operações morfológicas	Deconvolução	Métodos de registro	Ajuste de contraste
Trabalho proposto	X	X	X			X
Bisque	X					
Omero						
BioIMAX						
Slicer Drop	X					
BioSig3D	X					
BioimageXD	X	X	X	X	X	X
Vaa3D	X	X			X	X
ImageJ	X	X	X	X	X	X
Fiji	X	X	X	X	X	X
3D Slicer	X	X	X			X
ICY	X	X	X	X	X	X
CellProfiler	X	X	X			X

Os softwares de bioimagem na nuvem fornecem poucos algoritmos para processamento de imagens. Esses softwares foram desenvolvidos para um propósito específico, por exemplo, o Omero foca no armazenamento e gerenciamento dos experimentos e o Slicer Drop na visualização 3D.

A Tabela 4 apresenta uma comparação dos algoritmos de análise quantitativa do trabalho proposto com os trabalhos relacionados.

Tabela 4 – Comparação dos algoritmos de análise quantitativa dos trabalhos relacionados.

	Separação de objetos	Extração de características	Seleção de atributos	Aprendizado de máquina	Análise de colocalização
Trabalho proposto	X	X	X	X	
Bisque		X			
Omero					
BioIMAX		X			X
Slicer Drop					
BioSig3D	X	X			
BioimageXD	X	X			X
Vaa3D	X	X			
ImageJ	X	X			X
Fiji	X	X			X
3D Slicer	X	X			
ICY					
CellProfiler	X	X	X	X	

O ambiente proposto e o CellProfiler são os únicos trabalhos citados que fornecem funcionalidades de seleção de atributos e aprendizado de máquina. Na Tabela 4 é possível visualizar que grande parte dos softwares de biomagens não focam na análise quantitativa, necessitando que os cientistas utilizem outros softwares para a análise. O diferencial do trabalho proposto é que ele fornece o ambiente de visualização, processamento e análise quantitativa integrados, ou seja, após a extração dos atributos ele poderá inferir classificações sem a necessidade de trocar de ambiente.

A Tabela 5 apresenta uma comparação das arquiteturas e outras funcionalidades do trabalho proposto com os trabalhos relacionados.

Tabela 5 – Comparaçāo das arquiteturas e outras funcionalidades com os trabalhos relacionados.

	Arquitetura do processador	Sistema operacional	Linguagem de programação	Integração com outros ambientes	<i>Plug-ins</i>	Formatos suportados
Trabalho proposto	64 bits	Windows	Python	Sim	Não	+ de 100
Bisque	64 bits	Linux, iOS e Windows	Matlab, Python e Java	Sim	Não	+ de 100
Omero	64 bits	Linux, iOS e Windows	Python, Java e C++		Não	+ de 100
BioIMAX			Adobe Flex	Não	Não	
Slicer Drop		Linux, iOS e Windows	JavaScript	Não	Não	+ de 10
BioSig3D						
BioimageXD	64 bits	Linux, iOS e Windows	Python	Não	Sim	+ de 10
Vaa3D	64 bits	Linux, iOS e Windows	C++	Não	Sim	+ de 15
ImageJ	64 bits	Linux, iOS e Windows	Java	Não	Sim	+ de 100
Fiji	64 bits	Linux, iOS e Windows	Java	Não	Sim	+ de 100
3D Slicer	64 bits	Linux, iOS e Windows	Python	Não	Sim	+ de 30
ICY	64 bits	Linux, iOS e Windows	Java	Sim	Sim	+ de 100
CellProfiler	64 bits	Linux, iOS e Windows	Java	Sim	Não	+ de 100

A limitação do sistema operacional pode ser um fator que iniba a utilização do software de bioimagem. Na Tabela 5, os ambientes na nuvem podem ser utilizados em diversos sistemas operacionais que tenham suporte a um navegador web. Nessa tabela, o sistema operacional foi considerado para a instalação do ambiente, ou seja, se o ambiente for disponível na nuvem o sistema operacional listado é referente a compatibilidade para instalação do servidor. O servidor do ambiente proposto poderá ser compilado para Linux e iOS, pois ele utiliza linguagens e bibliotecas independentes do sistema operacional.

O softwares locais em sua maioria utilizam a arquitetura baseada em *plug-ins*, o que

não ocorre nos softwares na nuvem. O ImageJ tem sido a plataforma dominante, devido a sua arquitetura de *plug-ins*. Os pesquisadores escrevem centenas de *plug-ins* para processamento, visualização e análise de imagens em diversas áreas no ImageJ e disponibilizam esses métodos para a comunidade. Essa é uma das principais razões para disseminação e sucesso do software ImageJ (ELICEIRI et al., 2012). Assim, outros softwares locais, como, Vaa3D, ICY e BioImageXD também adotaram posteriormente a arquitetura baseada em *plug-ins*.

O ambiente proposto está bem posicionado em relação aos trabalhos relacionados. Os métodos fornecidos permitem os usuários explorarem as amostras, executarem algoritmos para segmentar uma região de interesse e extraírem dados quantitativos para a análise de uma amostra. A visualização volumétrica e extração de atributos em um objeto 3D possibilitam ao usuário identificar fenômenos que diversas vezes não é possível em um modelo 2D.

O BioSig3D é o ambiente que possui características similares ao ambiente proposto, porém ele não contempla algumas características devido ao seu objetivo ser diferente do BioWebVis. Além disso, ele redireciona o usuário para o ambiente ParaViewWeb para a visualização 3D, criando um arquivo de visualização. No ambiente proposto os módulos de segmentação, visualização e análise são totalmente integrados, o usuário não precisa trocar de ambiente.



# CAPÍTULO 4

## BioWebVis - Ambiente web para citomorfometria utilizando imagens 3D

Nesta seção será apresentado o BioWebVis, ambiente proposto neste trabalho para citomorfometria utilizando imagens 3D. Primeiramente, será apresentada uma visão geral do ambiente, com suas principais funcionalidades. Em seguida, serão apresentados detalhes da arquitetura do software e uma discussão das tecnologias envolvidas, que possibilitam o uso de técnicas de Visualização Computacional em um ambiente web.

### 4.1 Visão Geral

Nessa sessão serão apresentadas as funcionalidades do BioWebVis<sup>1</sup>. As funcionalidades do ambiente são divididas em 5 categorias: processamento de imagens, visualização 3D, citomorfometria, seleção de atributos e aprendizado de máquina. A Figura 12 apresenta o fluxo das atividades do BioWebVis.

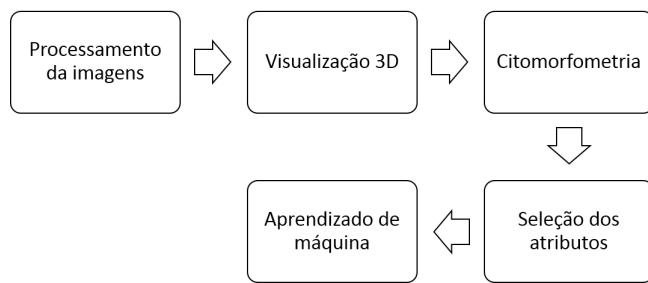


Figura 12 – Visão geral do BioWebVis.

Primeiramente, o usuário selecionará os algoritmos de processamento de imagens, tais como, redução de ruído e segmentação, com o objetivo de melhorar o conjunto de imagens antes da reconstrução 3D. Em seguida, ele selecionará o método de visualização 3D para

<sup>1</sup> Para mais detalhes, acesse o vídeo com a apresentação do ambiente no link <https://goo.gl/gUqvoa>.

a exploração científica da amostra em análise. Ao realizar a extração da isosuperfície, o usuário poderá submete-lá a análise citomorfometrica, em que serão extraídos 17 atributos. Durante a análise dos dados, o ambiente permite realizar a seleção dos atributos para reduzir a dimensionalidade e obter um melhor conjunto de atributos. Por fim, os cientistas poderão utilizar os algoritmos de aprendizado de máquina para auxilia-lós na descoberta de padrões, fornecendo assim informações relevantes para a tomada de decisão.

#### 4.1.1 Visualização 3D e Citomorfometria

O BioWebVis possibilita a qualquer usuário cadastrado o *upload* de conjuntos de dados, visualização e análise. O maior espaço da página principal é destinada à visualização, permitindo uma maior liberdade ao usuário ao interagir com a amostra em análise. As páginas web são adaptáveis à diferentes tamanhos de tela, em que a maior proporção da tela é também destinada a visualização. A Figura 13 apresenta a página de visualização 3D, e os menus de interação, análise e segmentação.

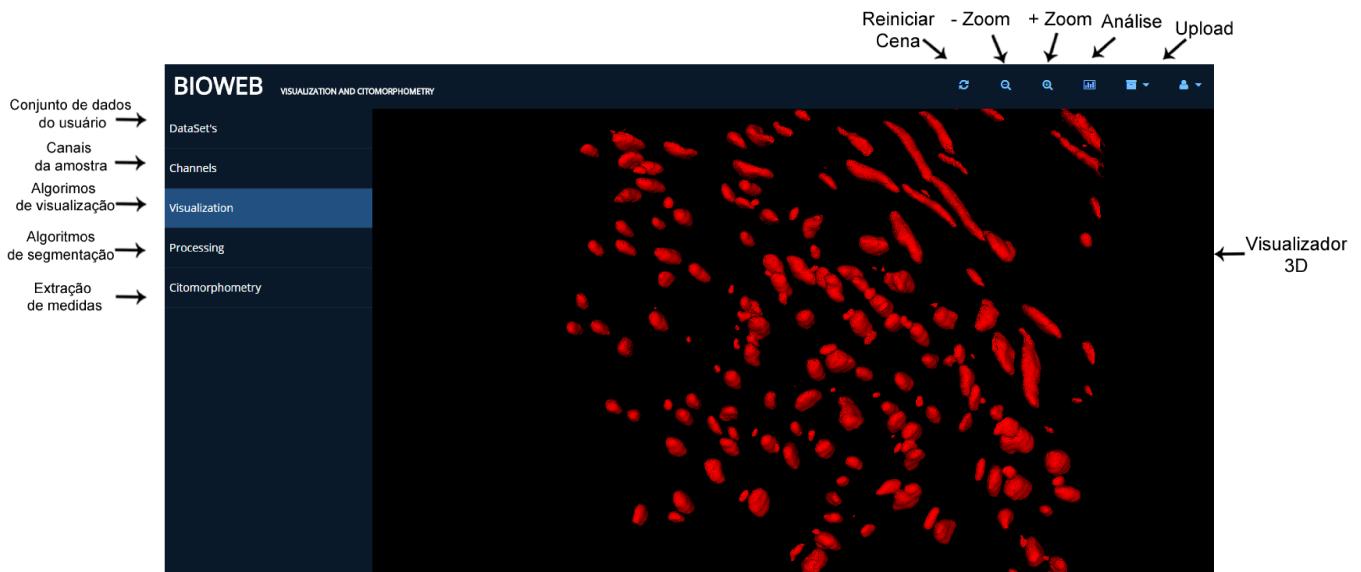


Figura 13 – Tela inicial do BioWebVis.

O ambiente reconhece mais de 100 formatos de imagem, pois utiliza o *framework* Bio-Formats, proposto por Allan et al. (2012). Após efetuar o *upload*, as amostras ficam disponíveis para visualização ou exclusão (e.g., Figura 14(a)). No envio das amostras é possível agrupá-las para uma melhor organização e predizer a classe da amostra, para compor a base de aprendizado dos algoritmos de classificação.

Os microscópicos de fluorescência fornecem imagens de multicanais. Na extração imagens de multicanais é utilizando diferentes tipos de fluorocromos (marcadores utilizados para destacar, por exemplo, proteínas ou organelas celulares). Cada fluorocromo re-

sulta em uma imagem, denominada como canal (CHITYALA; PUDIPEDDI, 2015). O ambiente permite a visualização de cada canal separadamente e também permite ao usuário selecionar os canais desejados para gerar a visualização 3D (e.g., Figura 14(b)). O pré-processamento, segmentação e análise citomorfométrica são realizados em cada canal separadamente.

Os resultados obtidos podem ser armazenados, exportados para um arquivo do tipo Valores Separados por Vírgulas (*Comma-separated values* – CSV) ou visualizados no próprio ambiente (e.g., Figura 14(c)) e analisados no módulo de análise e aprendizado de máquina.

Na reconstrução 3D pode-se utilizar os algoritmos *Volume Ray-casting* e *Marching Cubes* (e.g., Figura 14(d)). Ao renderizar uma amostra utilizando o *Marching Cubes* deverá ser informado o valor da variável *isovalue*. Por padrão, o BioWebVis fornece o valor 10 para o *isovalue* e também apresenta como sugestão o *isovalue* obtido pelo algoritmo de segmentação de Otsu (detalhes podem ser encontrados em (OTSU, 1979)), permitindo que o usuário escolha o valor apropriado para o seu experimento.

O usuário pode montar uma sequência de execução de algoritmos de pré-processamento e segmentação para o processar antes da reconstrução 3D (e.g., Figura 14(f)).

The figure consists of six panels labeled (a) through (f), each showing a different part of the BioWebVis software interface:

- (a) DataSet's:** A dropdown menu showing "B90-594A LAM1". Below it are "Open" and "Delete" buttons.
- (b) Channels:** Shows two channels selected: "Channel 0" (green) and "Channel 1" (red).
- (c) Details:** A table with columns: id, surface\_area, volume, NSI, curvature\_mean, and curvature\_std. Data rows are provided for entries 0 through 5, along with a mean row at the bottom.
- (d) Visualization:** Set to "Surface (marching cubes)". It shows an input field for "Iso-Value (0-255)" with "10" entered, and a dropdown for "Iso-Value with Otsu = 34".
- (e) Cytomorphometry:** Set to "Extract each cell separately". A list of checked metrics includes: surface\_area, volume, NSI, curvature\_mean, curvature\_std, curvature\_median, curvature\_entropy, and sphericity.
- (f) Processing:** A table with columns: seq, method, and param. It contains two rows: "1 threshold lower:0,upper:255" and "2 watershed level:0.5,minimum:0". Below the table are "checkmark", "+", and "eraser" buttons.

Figura 14 – Funcionalidades do BioWebVis para ViSC e Cytometria. (a) Seleção do conjunto de dados. (b) Escolha dos canais. (c) Visualização das medidas extraídas. (d) Seleção e configuração dos algoritmos de ViSC. (e) Extração das medidas utilizando métodos de citometria. (f) Pré-processamento e segmentação.

Na implementação dos algoritmos de processamento, foram utilizadas as bibliotecas

*SimpleITK*, *SciPy* e *VTK*. Os algoritmos disponibilizados na opção de menu referente a Figura 14(f) são:

- ❑ **Redução de Ruído:** Filtro de média (*SciPy* - *mean\_filter*), mediana (*SciPy* - *median\_filter*) e gaussiano (*SciPy* - *gaussian\_filter*) - detalhes podem ser encontrados em (GONZALEZ; WOODS, 2010).
- ❑ **Segmentação:** Limiarização (*SimpleIK* - *BinaryThresholdImageFilter*), Otsu (*SimpleIK* - *OtsuThresholdImageFilter*) e Watershed (*SimpleIK* - *MorphologicalWatershedImageFilter*) - detalhes podem ser encontrados em (GONZALEZ; WOODS, 2010).
- ❑ **Processamento morfológico:** Dilatação (*SimpleIK* - *BinaryDilateImageFilter*), erosão (*SimpleIK* - *BinaryErodeImageFilter*), abertura (*SimpleIK* - *BinaryOpeningByReconstructionImageFilter*) e fechamento (*SimpleIK* - *BinaryClosingByReconstructionImageFilter*) - detalhes podem ser encontrados em (GONZALEZ; WOODS, 2010).
- ❑ **Separação de objetos:** Separação de células com algoritmo Watershed (*SimpleITK* - *MorphologicalWatershedImageFilter* e *RelabelComponentImageFilter*) e rotulação de objetos 3D (*VTK* - *vtkPolyDataConnectivityFilter*) - detalhes podem ser encontrados em (SCHROEDER; LORENSEN; MARTIN, 2004).
- ❑ **Superfície 3D:** Reduzir o número de triangulações de uma malha, preservando sua topologia (*VTK* - *vtkDecimatePro* e *vtkSmoothPolyDataFilter*) - detalhes podem ser encontrados em (SCHROEDER; LORENSEN; MARTIN, 2004).

O BioWebVis realiza a extração de dados quantitativos utilizando a superfície obtida com o algoritmo *Marching Cubes* (e.g., Figura 14(d)). Na mensuração das características o usuário pode selecionar se deseja extrair os atributos de cada célula separadamente, ou se deseja extrair os atributos considerando toda a amostra. Se a extração for por célula, o filtro de conectividade é utilizado para separar os objetos e então cada célula do objeto é submetido às técnicas de citomorfometria. No ambiente são extraídas 17 medidas:

- ❑ Área da superfície;
- ❑ Volume, volume no eixo  $x$ , volume no eixo  $y$  e volume no eixo  $z$ ;
- ❑ Coeficientes  $k_x$ ,  $k_y$ , e  $k_z$
- ❑ Curvatura (média, mediana, desvio padrão, entropia);
- ❑ NSI e Esfericidade;
- ❑ Convexidade (por área da superfície e por volume);

- Diâmetro esférico equivalente.

O usuário pode extrair as medidas utilizando o processamento em lote. Nele é possível configurar os algoritmos de pré-processamento, segmentação e extração da superfície. Também é permitido selecionar quais canais e quais amostras serão processadas. Durante o processamento é apresentado uma barra de progresso e a cada amostra processada os atributos são armazenados na base de dados para uma análise posterior.

Os resultados obtidos podem ser armazenados, exportados para um arquivo CSV e visualizados no próprio ambiente (Figura 14(c)). Ao visualizar os resultados é possível renderizar a superfície do objeto clicando no identificador, e então o BioWebVis redireciona para a página de visualização do objeto de interesse, conforme mostra a Figura 15.

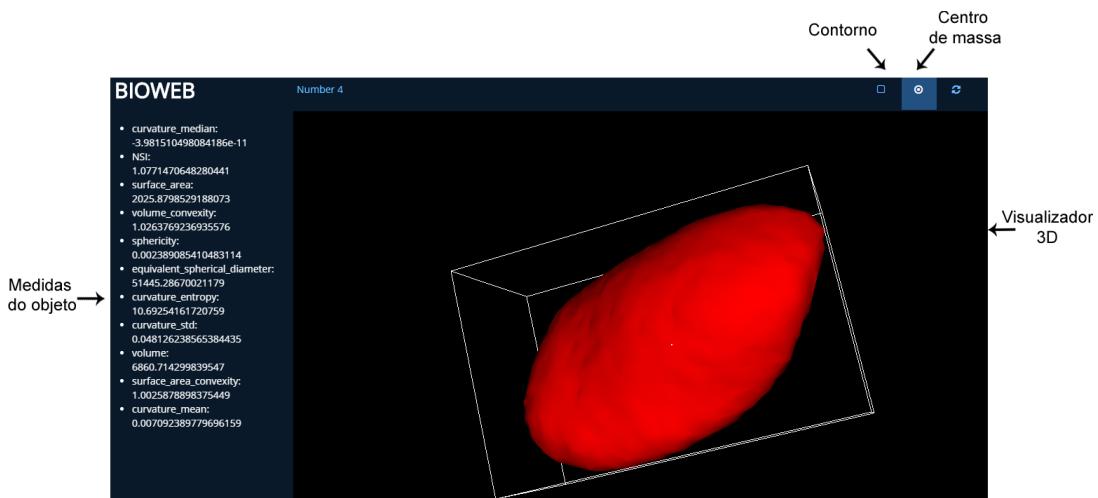


Figura 15 – Visualização de um objeto de interesse.

As interações com a cena são realizadas por meio da barra de ferramentas superior ou por meio do mouse. Na barra de ferramentas é possível aumentar e diminuir o zoom e reiniciar a câmera da cena em seu estado original. O visualizador 3D permite por meio do mouse visualizar a cena em diferentes perspectivas, aumentar e diminuir o zoom e selecionar um objeto.

Ao visualizar uma superfície é possível selecionar um objeto de interesse de duas formas: pelas coordenadas  $x$  e  $y$  obtidas por um clique duplo ou pela extração das medidas de citomorfometria. O algoritmo de extração de medidas baseia-se na conectividade para a separação dos objetos, cada objeto extraído recebe um identificador que pode ser utilizado para visualizar o objeto de interesse posteriormente. Após encontrar o objeto de interesse, o BioWebVis redireciona o usuário para uma nova página, conforme a Figura 15.

Na visualização do objeto de interesse todas as medidas são apresentadas na barra lateral, além de interações com o objeto é possível adicionar um delimitador de fronteira e um indicador do centro de massa do objeto. Uma amostra é composta por diversas células,

a visualização de um objeto de interesse permite ao usuário explorar a estrutura de uma célula, possibilitando analisar as deformações e as medidas fornecidas pelo ambiente.

#### 4.1.2 Análise e Aprendizado de Máquina

O módulo de análise permite ao usuário realizar testes estatísticos e inferir classificações nas amostras que foram extraídos os atributos morfológicos. Após a extração dos atributos, os dados quantitativos ficam disponíveis para a análise. O usuário selecionará quais grupos de amostras serão carregadas para a análise e então poderá ser realizado diversos testes, que serão descritos a seguir.

Os atributos fornecidos pelo BioWebVis estão descritos em diferentes escalas de mensuração, por exemplo, área da superfície e índices de curvatura. Para normalizar os dados brutos é fornecido o método Z-Score, possibilitando que o usuário faça seus experimentos em uma escala normalizada.

O ambiente fornece 17 atributos, porém alguns desses atributos podem influenciar negativamente na inferência de conhecimento, pois podem estar, por exemplo, altamente correlacionados com outros atributos. Reduzir a dimensionalidade dos atributos pode melhorar a eficácia dos algoritmos de aprendizado de máquina, simplificar o modelo e reduzir o tamanho dos dados brutos. O ambiente fornece gráficos de dispersão (e.g. Figura 16(a)), gráficos de distribuição normal (e.g. Figura 16(b)), análise de regressão (e.g. Figura 16(c)), análise de variância (e.g. Figura 16(d)) e coeficientes de Pearson.

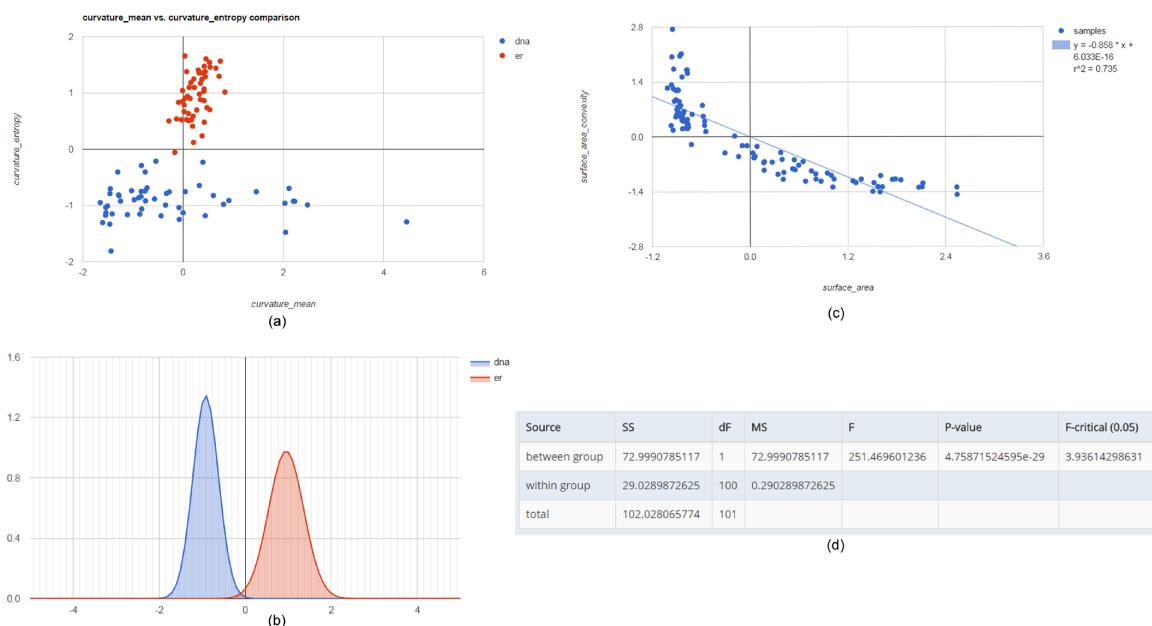


Figura 16 – Funcionalidades para análise dos atributos. (a) Gráfico de dispersão. (b) Gráfico de distribuição normal. (c) Análise de regressão. (d) Análise de variância entre (*between group*) e dentro (*within group*) dos grupos.

O gráfico de dispersão é útil para analisar se dois atributos conseguem separar as amostras em suas respectivas classes utilizando as medidas extraídas. O gráfico de distribuição normal utiliza a média e desvio padrão das medidas extraídas para gerar uma curva. A área da curva é expressa pela quantidade de desvios padrões de distância da média (topo da curva) (SEABRA, 2015). Com esse gráfico o usuário pode analisar o comportamento de um atributo sobre as classes das amostras e se as curvas das classes estão sobrepostas. A análise de regressão permite investigar a relação entre dois atributos, é fornecido o grau de correlação linear entre dois atributos utilizando o coeficiente de correlação de Pearson. Se o coeficiente de correlação é igual a 0, os dois atributos não dependem linearmente, se igual a 1 existe uma correlação linear positiva, se igual -1 existe uma correlação linear negativa (LIU; MOTODA, 2007; HALL, 1999). Análise de variância permite comparar a variação de um atributo entre as classes e dentro das classes. Um bom atributo deve variar entre as classes e ter uma baixa variação dentro das classes. O BioWebVis também permite a redução da dimensionalidade utilizando o algoritmo PCA e com o *Extra Tree*. Na implementação do PCA e do *ExtraTree* foi utilizou a biblioteca *Scikit-Learn* com os métodos *PCA* e *ExtraTreesClassifier*.

Após a aplicação dos métodos estatísticos e seleção dos atributos, o usuário poderá inferir classificações nas amostras e validar os experimentos realizados. Ao realizar o envio das amostras para o ambiente, o especialista pode informar a classe da amostra, para utilizar como base de treinamento. Na implementação dos classificadores foi utilizado a biblioteca *Scikit-Learn*. O ambiente disponibiliza os seguintes classificadores supervisionados: o Naive Bayes (*Scikit-Learn - GaussianNB*) e o QDA (*Scikit-Learn - QuadraticDiscriminantAnalysis*) (classificador baseado na teoria de decisão de Bayes), o SVM (*Scikit-Learn - SVC*) (classificador não probabilístico linear), o K-NN (*Scikit-Learn - KNeighborsClassifier*) (classificador baseado em instâncias que utiliza a distância entre as amostras) e o Multilayer Perceptron (*Scikit-Learn - MLPClassifier*) (classificador baseado em redes neurais). E também fornece o *K-means* (*Scikit-Learn - KMeans*) como algoritmo de aprendizado não supervisionado.

Utilizando os métodos de classificação, o usuário poderá inferir a classificação de uma nova amostra ou também validar o modelo proposto. Para validar o modelo proposto, o ambiente fornece o método de validação cruzada *K-Fold*. E também fornece a acurácia, precisão, *F1-Score* e matriz de confusão para validar os resultados dos algoritmos supervisionados. Para validar os resultados do algoritmo não supervisionado é disponibilizado a medida *V-measure*.

## 4.2 Arquitetura

Segundo (JOURDAIN; AYACHIT; GEVECI, 2010), algumas aplicações de visualização na web fazem a renderização dos objetos no lado do cliente e tornam-se ineficientes

quando grandes conjuntos de dados são renderizados, por exemplo, *smartphones* possuem poder computacional limitado. A arquitetura do BioWebVis foi baseada no modelo cliente/servidor em que todo processamento é realizado nos servidores.

Nesse modelo o processamento dos algoritmos de análise e visualização são realizados no servidor. A Figura 17 ilustra a arquitetura proposta para o ambiente BioWebVis. Essa arquitetura foi baseada no ParaViewWeb, devido a capacidade de processar grandes conjuntos de dados (JOURDAIN; AYACHIT; GEVECI, 2010). Ela é composta por três aplicações: cliente, *VTKServer* e *APPServer*.

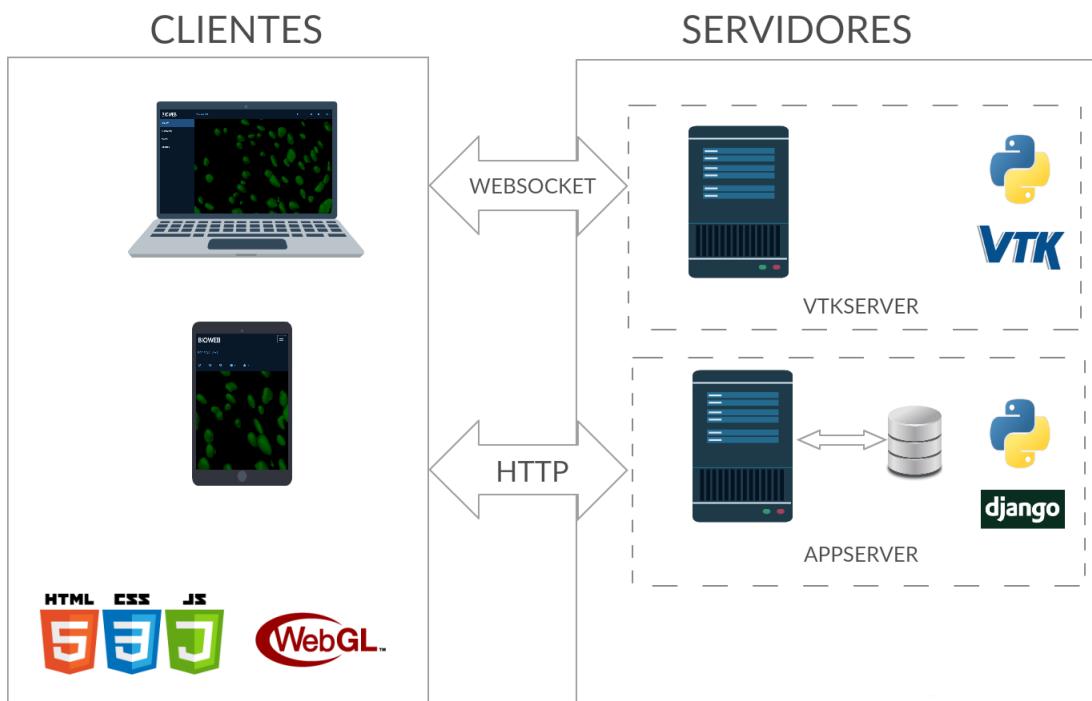


Figura 17 – Arquitetura do ambiente BioWebVis.

O cliente realiza via protocolo *websocket* solicitações de cenas 3D ou execução de algoritmos ao *VTKServer* e o *APPServer* gerencia os conjuntos de dados, os usuários do ambiente e a análise dos atributos extraídos.

A aplicação cliente é responsável pela interação entre os usuários e os servidores por meio de páginas web. No desenvolvimento das páginas foram utilizados: Linguagem de Marcação de Hipertexto (*HyperText Markup Language 5* – HTML 5), Folhas de Estilo em Cascata 3 (*Cascading Style Sheets 3* – CSS 3) e JavaScript. A biblioteca VTKWeb escrita em JavaScript auxilia na comunicação *WebSocket* com o servidor *VTKServer*, bem como na renderização de objetos 3D baseado na especificação WebGL do HTML 5. O cliente pode ser qualquer dispositivo que possui um navegador web atualizado com suporte ao WebGL, tais como, computadores de mesa, *smartphones* e *tablets*.

O *APPServer* é uma aplicação escrita em Python com Django (biblioteca para construção de aplicações web) responsável por gerenciar o *website*. As principais funcionalidades

do *APPServer* são:

- ❑ **Disponibilização:** disponibilização das páginas web para o acesso dos usuários;
- ❑ **Autenticar:** gerenciar cadastro e autenticação dos usuários;
- ❑ **Gerenciar os conjuntos de dados:** realizar *uploads*, armazenagem e organização em grupos dos conjuntos de amostras;
- ❑ **Gerenciar resultados:** armazenar as medidas extraídas.
- ❑ **Análise estatística:** fornecer métodos para a análise estatística dos atributos extraídos.
- ❑ **Aprendizado de máquina:** fornecer métodos de ML para inferir classificações.

O *VTKServer* é uma aplicação escrita em Python responsável por realizar o pré-processamento, segmentação, as análises citomorfométricas e aplicar os algoritmos de visualização 3D com o *Visualization Toolkit* (VTK).

A comunicação entre o cliente e o *VTKServer* é realizada por meio do protocolo *WebSocket*. O *WebSocket* é um protocolo para a comunicação bidirecional, em que mensagens podem ser trocadas entre clientes e servidores reduzindo a complexidade sobre a comunicação em tempo real. O *WebSocket* demonstra-se ser eficiente em aplicações de tempo real, reduzindo a sobrecarga e fornecendo uma comunicação eficiente (FETTE; MELNIKOV, 2011). No trabalho proposto por (MARION; JOMIER, 2012), o protocolo *WebSocket* em comparação com o Protocolo de Transferência de Hipertexto (*Hypertext Transfer Protocol* – HTTP) apresenta melhores resultados.

Por meio de um navegador o usuário realizará a autenticação no *APPServer* com um e-mail e senha. Após a autenticação será retornado a identificação do usuário e as identificações dos conjuntos de dados para o cliente. O usuário pode solicitar a renderização de um conjunto de dados para o *VTKServer*, que processará as informações e enviará vários fluxos de vértices (primitivas) ao cliente. Em seguida o usuário poderá solicitar a análise citomorfometrica da cena renderizada.

Com *WebSocket* é utilizada a estratégia de carregamento progressivo, ou seja, quando um cliente solicita um objeto 3D o *VTKServer* envia as primitivas em pequenos blocos, o cliente vai recebendo os blocos e apresentando para o usuário. Isso evita que a biblioteca JavaScript no lado do cliente possa sobrecarregar e não apresentar corretamente ao usuário o modelo 3D.

Os servidores podem ser computadores ou supercomputadores que irão responder a solicitações dos clientes, na arquitetura em nuvem, o número de servidores pode ser escalável de acordo com a necessidade de processamento.

Na próxima seção serão apresentadas algumas análises que ajudaram na escolha das tecnologias adotadas no BioWebVis, e que contribuiriam para a definição da arquitetura.

## 4.3 Visualização Usando WebGL

Esta seção apresenta os resultados obtidos ao analisar diferentes bibliotecas para visualização WebGL, em que o quesito desempenho e integração com o VTK foi fundamental para determinar as bibliotecas e a arquitetura do BioWebVis.

O VTK é uma biblioteca *open-source* para processamento e visualização 3D utilizada em diversos segmentos, tais como, visualização médica, industrial e biológica (SCHROEDER; LORENSEN; MARTIN, 2004). Nos softwares para visualização e processamento de imagens biológicas, o VTK demonstrou-se eficaz na manipulação de objetos 3D obtidos a partir de microscópios, sendo utilizado em ferramentas como o BioImageXD e ICY (KAN-KAANPÄÄ et al., 2012; CHAUMONT et al., 2012). Essa biblioteca foi utilizada para a reconstrução de objetos 3D no BioWebVis.

### 4.3.1 Bibliotecas WebGL

Com o potencial do WebGL, diversas bibliotecas em JavaScript foram propostas para auxiliar a manipulação de objetos 3D no navegador. As bibliotecas X Toolkit (XTK) (HAEHN et al., 2014), THREE.js (CABELLO, 2010), X3DOM (BEHR et al., 2009) e VTKWeb (SCHROEDER; LORENSEN; MARTIN, 2004) foram analisadas com base em quatro atributos: desempenho, confiabilidade, interação com o usuário e integração com o VTK.

O XTK é uma biblioteca JavaScript para ViSC usando HTML 5 e WebGL. Ela é utilizada em aplicações de VI focadas em imagens médicas, como neuroimagens, e permite interações via mouse e teclado. Para integração com outras aplicações, o XTK aceita os seguintes formatos de arquivos: DICOM, STL e VTK. Essa biblioteca foi utilizada no ambiente Slicer Drop, proposto por Haehn et al. (2014).

O THREE.js é uma biblioteca JavaScript para criação de objetos 3D com WebGL. No trabalho proposto por Jacinto et al. (2012), o THREE.js foi utilizado em conjunto com o VTK para visualização e segmentação de imagens médicas por meio do envio de modelos 3D no formato OpenCTM. O THREE.js também permite formatos como STL e VTK. Essa biblioteca foi utilizada no ambiente Bisque, proposto por (KVILEKVAL et al., 2010).

O X3D é um padrão livre para representação de cenas 3D baseada em Linguagem de Marcação Extensível (*eXtensible Markup Language – XML*). O X3DOM é um projeto de integração do X3D no HTML 5, permitindo que páginas web possam renderizar uma cena em X3D. No trabalho proposto por Congote (2012) foi adicionado propriedades ao X3DOM para visualização volumétrica em imagens médicas no formato DICOM ou com imagens 2D fatiadas (BEHR et al., 2009).

O VTKWeb é uma extensão do VTK em JavaScript que auxilia a comunicação entre um cliente e um servidor VTK. O servidor deve realizar o envio de um fluxo de

imagens ou primitivas WebGL (sequência de vértices) que representa o objeto 3D a ser renderizado (SCHROEDER; LORENSEN; MARTIN, 2004). Essa biblioteca é baseada no *framework* ParaViewWeb, proposto por Jourdain, Ayachit e Geveci (2010).

### 4.3.2 Avaliação das Bibliotecas

Para escolher o melhor modelo de comunicação com o cliente baseando-se no atributo desempenho, foram selecionadas quatro amostras volumétricas de 647x647 de diferentes tamanhos no eixo *z*. A integração do VTK no lado servidor com as bibliotecas do lado cliente foi realizado por meio dos formatos de arquivos: VTK, STL, X3D e OpenCTM. Eles representam um modelo 3D obtidos com o algoritmo *marching cubes* das quatro amostras. A Tabela 6 apresenta o modelo 3D das amostras e seus respectivos tamanhos em *kilobyte*.

Tabela 6 – Tamanhos de diferentes tipos de arquivos para representação 3D.

	Tamanho do eixo Z	VTK (kb)	STL (kb)	X3D (kb)	OpenCTM (kb)
Amostra 1	47	12,812	42,200	20,385	820
Amostra 2	33	2,943	10,048	4,750	209
Amostra 3	66	5,367	18,151	8,644	820
Amostra 4	219	48,209	152,591	76,447	3,010SS

O formato OpenCTM permitiu armazenar um modelo 3D em uma quantidade de bytes bem inferior aos demais formatos. No tráfego de informações via internet, quanto menor a quantidade dados transmitidos melhor é o tempo de resposta para o usuário. O trabalho BioSig3D proposto por Bilgin et al. (2016), utiliza o formato VTP para trafegar o objeto 3D para o ambiente de visualização. Esse formato é similar ao formato VTK.

Para analisar o desempenho foi ponderado o tempo gasto em reconstruir as superfícies das amostras, em criar o arquivo 3D e em apresentar via WebGL a superfície para o usuário no navegador. O tempo gasto na criação dos arquivos é importante, pois alguns formatos necessitam de um tempo maior para a sua criação em comparação à outros formatos.

Os testes foram realizados com quatro amostras de diferentes tamanhos de arquivos. A Tabela 7 apresenta os resultados dos testes expresso em segundos.

Durante a análise de desempenho as bibliotecas XTK e Three.js apresentaram erros por não conseguirem renderizar grandes conjuntos de dados, pois o processamento dos objetos 3D são realizados no cliente. Jourdain, Ayachit e Geveci (2010) mencionaram que algumas aplicações de visualização na web são ineficientes quando grandes conjuntos de dados são renderizados. O Slicer Drop utiliza a biblioteca XTK e apresentou erros

Tabela 7 – Desempenho das bibliotecas de visualização com WebGL.

Biblioteca	Formato	Amostra 1 (seg.)	Amostra 2 (seg.)	Amostra 3 (seg.)	Amostra 4 (seg.)
XTK	VTK	7,08	2,74	5,12	Com erros
	STL	11,62	3,48	9,05	Com erros
Three.js	VTK	5,97	2,24	5,56	28,73
	STL	Com erros	4,46	9,82	Com erros
	OpenCTM	6,17	2,05	5,11	25,35
X3Dom	X3D	13,19	3,49	8,41	88,16
VTKWeb	Primitivas	<b>5,12</b>	<b>2,04</b>	<b>4,98</b>	<b>24,50</b>

ao renderizar grandes conjuntos de dados. O X3D não apresentou erros, porém foi a biblioteca com o pior desempenho.

A biblioteca VTKWeb obteve o melhor desempenho em todas amostras, conforme destacado em negrito na Tabela 7 e em grandes conjuntos de dados não gerou erros na apresentação do objeto 3D para o usuário.

O formato OpenCTM gerou os menores arquivos, porém durante a análise de desempenho ele não apresentou os melhores resultados, pois o VTK não consegue exportar uma superfície para o formato OpenCTM. Neste caso, primeiramente deve-se exportar para outro formato, como o STL, e em seguida utilizar uma biblioteca externa para converter de STL para OpenCTM.

O resultado final da análise foi realizada observando as médias de desempenho nas quatro amostras. A Figura 18 apresenta um gráfico com as médias de cada biblioteca e o formato utilizado. Foram ignoradas as análises que apresentaram erros.



Figura 18 – Gráfico de desempenho das bibliotecas para manipulação do WebGL.

O VTKWeb apresentou o melhor desempenho, além de ser capaz de apresentar grandes conjuntos de dados sem erros. Com o VTKWeb elimina-se a tarefa de exportar o objeto 3D

para um arquivo, como ocorre no BioSig3D. Assim, o VTKWeb se mostrou ser confiável por não apresentar erros e permitiu interações entre o usuário e o objeto.



# CAPÍTULO 5

## Utilização de citomorfometria para classificação da localização de padrões subcelulares em isosuperfícies 3D

Nessa seção será apresentado um estudo de caso de caso em que o BioWebVis foi utilizado em todo o *pipeline* para extração de características morfológicas e localização dos padrões subcelulares em imagens 3D obtidas por meio da reconstrução de isosuperfícies.

### 5.1 Introdução

A proteômica é o estudo de um conjunto de proteínas de uma organela subcelular, tecido ou organismo. Elas são responsáveis pelo controle de quase todos os processos biológicos (BARBOSA et al., 2012). A localização subcelular de proteínas é essencial para a construção de modelos que capturem e simulem o comportamento celular, sendo útil para o diagnóstico precoce de doenças.

A localização de uma proteína pode auxiliar os cientistas a descrever as funções realizadas por ela, por exemplo, uma proteína localizada na membrana plasmática pode ter a função de transportador ou de canal iônico, por outro lado uma proteína citoplasmática pode ter a função de catalisador (CHEN; MURPHY, 2004).

Na patologia, alterações morfológicas nas estruturas celulares são importantes no auxílio dos métodos de diagnósticos. As características morfológicas das proteínas podem fornecer informações relevantes para reconhecer os padrões subcelulares e auxiliar os biólogos na predição dos comportamentos celulares.

O estudo de caso apresentado neste capítulo propõe uma metodologia para extração de características morfológicas dos padrões subcelulares em imagens 3D obtidas por meio da reconstrução de isosuperfícies. Classificadores foram utilizados para aprender os padrões subcelulares e fornecer a capacidade de distingui-los. Ao final, foi calculada a acurácia e os resultados foram comparados com outras abordagens.

### 5.1.1 Trabalhos relacionados

A microscopia de fluorescência é utilizada para destacar a distribuição espacial de cada proteína. Em muitas vezes a inspeção visual de uma imagem fornecida por esse tipo de microscópico não é suficiente para caracterizar a localização de um padrão subcelular. Assim, existe a necessidade de uma variedade de métodos computacionais para realizar a localização de padrões subcelulares, várias abordagens foram propostas para a predição da localização dos padrões.

Os métodos propostos utilizam a base de dados de células cancerígenas HeLa proposta por Velliste e Murphy (2002). HeLa é um tumor maligno do colo uterino amplamente utilizada em pesquisas científicas (GUILHERMANO et al., 2010). Para fornecer informações relevantes para distinguir os padrões subcelulares, foram desenvolvidas metodologias que utilizam os voxels dos volumes 3D para a extração de atributos, tais como, atributos de textura, atributos de distância e atributos morfológicos .

Velliste e Murphy (2002) propuseram uma metodologia para a localização de 10 diferentes padrões subcelulares em células HeLa obtidas por microscopia confocal de fluorescência. Usando algoritmos de ML foi possível realizar previsões dos padrões subcelulares. No trabalho proposto pelos autores, 28 atributos morfológicos e de distância foram extraídos para a validação do modelo proposto. Os autores utilizaram a referência do DNA na célula para a geração de atributos baseados em distância, tal como, a distância do centro da proteína até o centro do DNA. Com o algoritmo de classificação *Back-Propagation Neural Network* (BPNN) foi obtido 91% de acurácia. No trabalho proposto por Huang e Murphy (2004b), utilizando os mesmos 28 atributos, 9 foram selecionados pelo método Análise Discriminante Gradativa (*Stepwise Discriminant Analysis*) (SDA). Com o algoritmo de classificação *Majority Voting* foi obtido 96% de acurácia no modelo proposto.

Os atributos de distância que utilizam a referência do DNA podem ser problemáticos, pois algumas imagens podem não conter o DNA. As imagens podem ser extraídas por diferentes especialistas em diferentes laboratórios, então nem sempre uma imagem poderá ter o DNA. No trabalho proposto por Huang e Murphy (2004a), foram utilizados 14 atributos morfológicos e de distância (os atributos de distância que utilizam o DNA foram eliminados). Com o algoritmo SVM para a classificação das amostras, os autores obtiveram 89,1% de acurácia no modelo proposto.

No trabalho desenvolvido por Chen e Murphy (2004) foram extraídos 48 atributos, sendo eles: atributos de textura de Halarick, atributos de borda e atributos morfológicos. Os autores utilizaram o SDA para selecionar um conjunto de atributos que melhor discriminam as classes. Dos 48 atributos, foram selecionados 7 atributos, sendo um atributo de borda, cinco atributos de textura e o desvio padrão do volume como o único atributo morfológico. Com o algoritmo BPNN para a classificação das amostras, os autores obtiveram 98% de acurácia no modelo proposto.

A localização subcelular de proteínas em imagens 3D apresentou resultados promisso-

res em relação a imagens 2D (HUANG; MURPHY, 2004b). Os métodos descritos nessa seção diferem do método proposto, pois o presente trabalho utiliza a reconstrução de isosuperfície para a extração dos atributos e é utilizado somente atributos morfológicos.

## 5.2 Materiais e Métodos

Nesta seção serão descritos os materiais e métodos utilizados para o desenvolvimento e obtenção dos resultados.

### 5.2.1 Base de Dados

Para avaliar a metodologia proposta, foi utilizada uma base de imagens 3D de células HeLa proposta por Velliste e Murphy (2002). A base de dados é composta por 454 imagens 3D (aproximadamente 50 imagens por classe), cada volume possui um único componente celular. O volume 3D possui entre 14 e 24 fatias 2D, a dimensão dos voxels nas coordenadas  $x$ ,  $y$  e  $z$  foram  $0,049 \times 0,049 \times 0,203 \mu\text{m}$  (VELLISTE; MURPHY, 2002). As imagens foram capturadas utilizando um microscópico confocal de fluorescência, focalizando na localização de proteínas subcelulares e organelas (classes), como: *Retículo Endoplasmático* (RE), duas proteínas de Golgi (*giantin* e *gpp130*), LAMP2 (lisossomos), mitocôndria, nucleolina, actina, tubulina e o DNA (VELLISTE; MURPHY, 2002).

A Figura 19 apresenta, para cada classe, uma imagem (uma fatia do volume 3D).

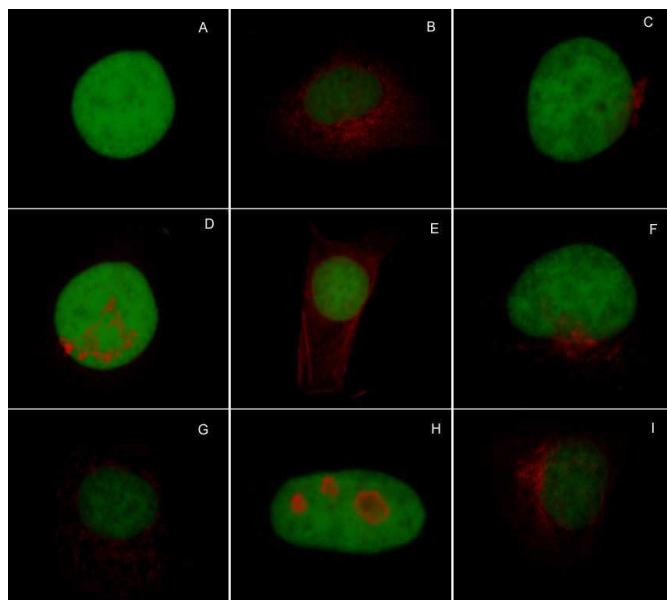


Figura 19 – Visualização da base de dados utilizando o software BioWebVis. (A) DNA, (B) RE, (C) Giantin, (D) GPP130, (E) F-Actina, (F) LAMP2, (G) Mitocôndria, (H) Nucleolina , (I) Tubulina.

Na Figura 19, dois canais foram destacados. O canal verde representa o DNA e o vermelho as proteínas ou organelas. Nessas imagens é possível localizar os padrões subcelulares em referência ao DNA. As proteínas de *giantin* e *gpp130* foram incluídas para avaliar a sensibilidade do conjunto de características extraídas, pois existe uma dificuldade em distingui-las em uma inspeção visual.

### 5.2.2 Metodologia Proposta

As imagens foram analisadas utilizando o ambiente BioWebVis. Vários algoritmos e técnicas disponíveis no ambiente foram utilizados (i) processamento de imagens - pré-processamento, segmentação, extração de características; (ii) visualização científica - DVR, reconstrução de isosuperfície; e (iii) reconhecimento de padrões - K-NN, SVM, QDA. O *pipeline* geral para análise dos dados é mostrado na Figura 20.

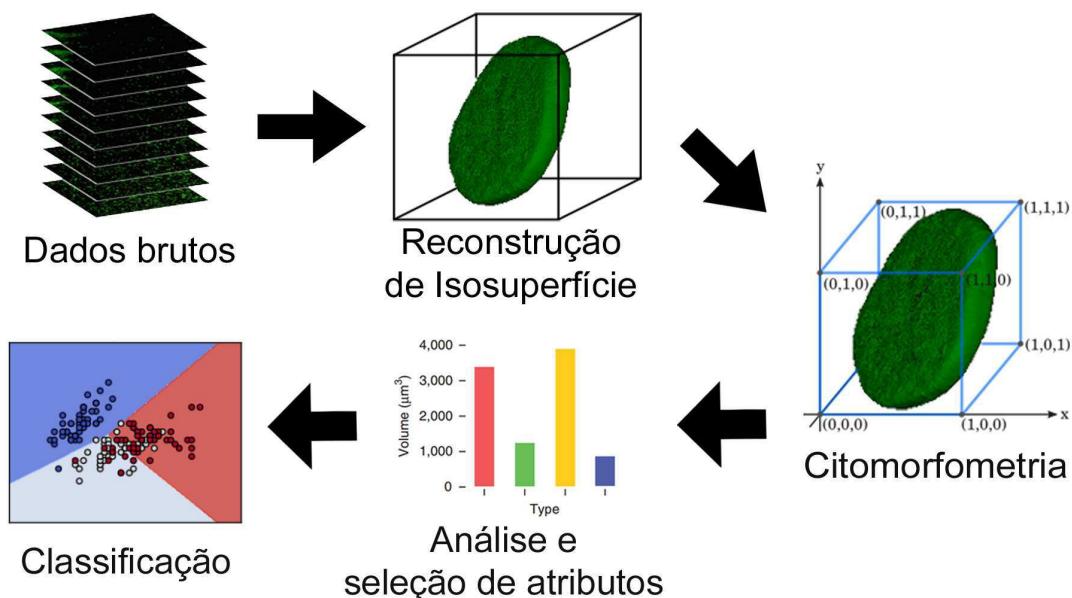


Figura 20 – *Pipeline* geral utilizado para análise dos padrões subcelulares.

Primeiramente, os dados brutos (um conjunto de imagens 2D) foram reconstruídos utilizando o algoritmo *Marching Cubes*. Depois da reconstrução cada proteína subcelular ou organela é delimitada por uma superfície externa. As superfícies extraídas são compostas por uma malha poligonal, permitindo obter informações quantitativas, como por exemplo, a área da superfície. Para todas as imagens, o *isovalue* escolhido foi 10 (o *isovalue* é um limiar aplicado sobre as intensidades dos voxels de modo a determinar a localização da superfície dentro do volume). A escolha do *isovalue* influencia na isosuperfície extraída, a Figura 21 apresenta duas isosuperfícies extraídas de um mesmo volume com limiares diferentes. O *isovalue* igual a 10 foi determinado com base em testes realizados. Inicialmente, o limiar foi obtido automaticamente com o método de Otsu, na

maioria das amostras o limiar obtido foi sempre maior que 10. Com esse limiar algumas regiões das imagens das classes LAMP2 e Mitocôndria foram perdidas devido a intensidade de seus voxels (Figura 21(B)). O limiar foi ajustado até 10, valor em que se obteve os melhores resultados.

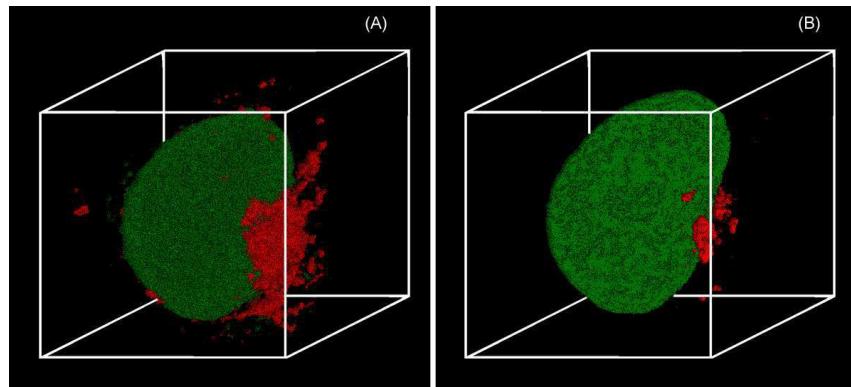


Figura 21 – Visualização da isosuperfície da classe LAMP2 utilizando o BioWebVis. (A) LAMP2 com *isovalue* igual a 10, (B) LAMP2 com limiar obtido pelo método de Otsu.

No próximo passo, para cada proteína/organela 17 atributos foram extraídos: área da superfície, volume, volume no eixo  $x$ , volume no eixo  $y$  e volume no eixo  $z$ , curvatura (média, mediana, desvio padrão, entropia), NSI, esfericidade, convexidade (por área da superfície e por volume), coeficientes  $k_x$ ,  $k_y$ , e  $k_z$  e diâmetro esférico equivalente.

Os atributos extraídos de cada proteína/organela estão em escalas diferentes. O *Z-Score* disponível no BioWebVis foi utilizado no processo de normalização para criar uma escala comum.

Após a normalização, os dados compartilham uma escala comum, que é importante para o reconhecimento de padrões. O próximo passo consiste em realizar a seleção dos atributos, processo para encontrar o conjunto de características que melhor separe as classes. No BioWebVis as seguintes técnicas estão disponíveis: análise de variância, análise de regressão, Correlação de Pearson e Árvore de Decisão. Todas essas técnicas foram utilizadas nesse estudo de caso para encontrar o melhor conjunto de características que representam cada classe (LIU; MOTODA, 2007).

Em seguida, algoritmos de aprendizado de máquina Naive Bayes, QDA, SVM, K-NN e *Multilayer Perceptron* disponíveis no BioWebVis foram utilizados para classificar as amostras e encontrar padrões nos dados. Para avaliar o desempenho dos algoritmos foi utilizado o método de validação cruzada *k-fold* obtendo-se a acurácia e o *F-Score*.

## 5.3 Resultados

A Figura 22 mostra exemplos de todas as isosuperfícies das classes da base de dados. O canal verde representa o DNA e o vermelho as proteínas ou organelas, assim é possível

visualizar a localização subcelular em torno do DNA. As medidas foram extraídas das superfícies de cada padrão subcelular e do DNA.

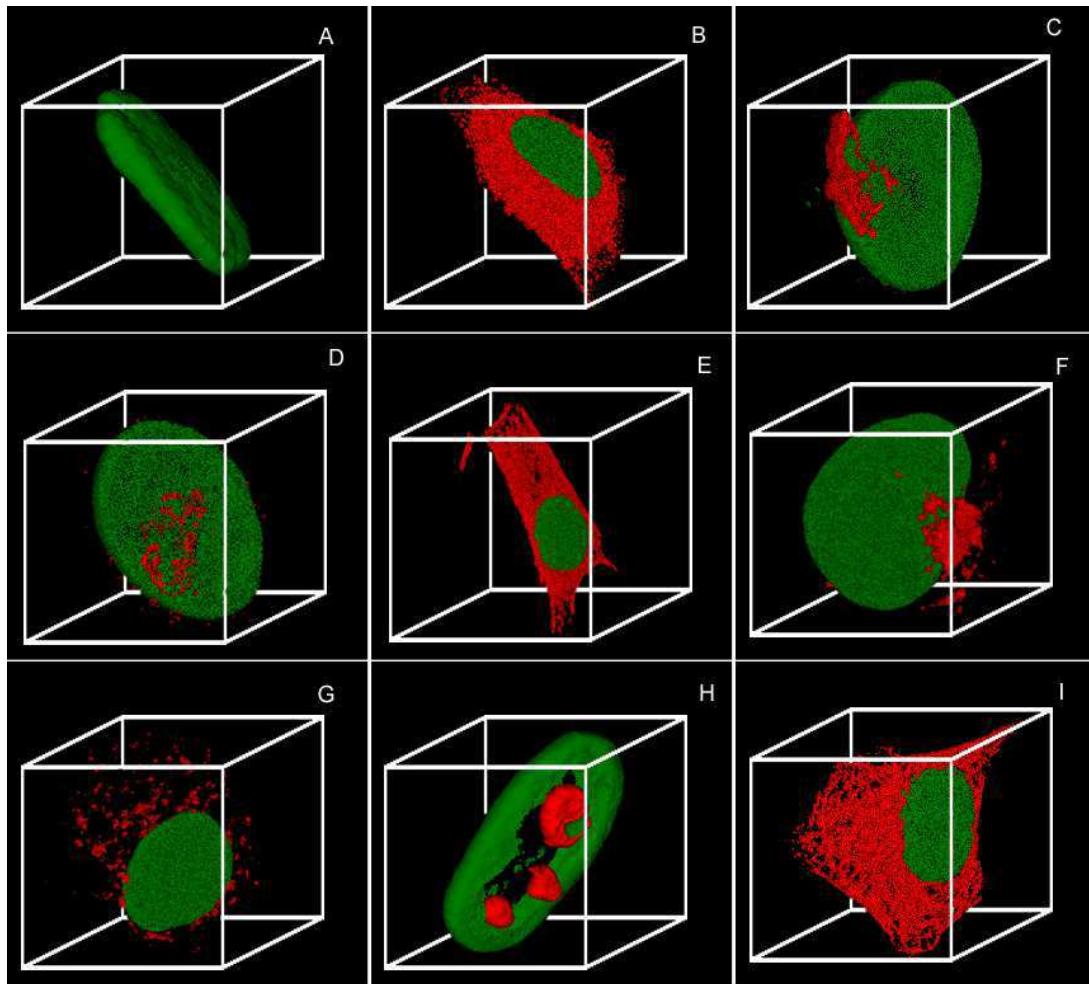


Figura 22 – Visualização das isosuperfícies utilizando o BioWebVis. (A) DNA, (B) RE, (C) Giantin, (D) GPP130, (E) F-Actina, (F) LAMP2, (G) Mitocôndria, (H) Nucleolina , (I) Tubulina.

Para os teste, os dados de entrada foram normalizados com Z-Score. Para obtenção dos resultados foi utilizado a validação cruzada  $k$ -fold com  $k = 10$ . Foi calculado o valor médio da acurácia e do  $F1$ -score para cada classificador. No classificador K-NN utilizou-se  $k = 1$ , pois apresentou os melhores resultados, no classificador *Multilayer Perceptron* utilizou-se 100 neurônios em camadas ocultas e no SVM utilizou o *kernel* linear. A Tabela 8 apresenta o resultado da classificação com todos os 17 atributos extraídos utilizando a citomorfometria.

Tabela 8 – Resultado dos classificadores com os todos os 17 atributos.

Classificador	<i>F1-score</i>	Acurácia	Tempo (seg.)
K-NN (k= 1)	0,9223	92,31%	1,33
Multilayer Perceptron	0,9305	93,41%	13,21
Naive Bayes	0,9439	94,53%	1,46
QDA	0,9391	94,10%	2,07
<b>SVM</b>	<b>0,9445</b>	<b>94,51%</b>	<b>2,25</b>

O SVM apresentou o melhor resultado, com acurácia de 94,51% e *F1-score* de 0,9445. Para reduzir a dimensionalidade dos atributos e otimizar a classificação foi utilizado a árvore de decisão *Extra Tree*, disponível na biblioteca *Scikit-Learn*. O *Extra Tree* consiste em randomizar a escolha do atributo para criar um ponto de corte ao dividir um nó de uma árvore e gerar várias árvores randômicas, os melhores pontos de cortes são escolhidos como regras das divisões (GEURTS; ERNST; WEHENKEL, 2006). Foram selecionados automaticamente 7 atributos: média, desvio padrão e entropia da curvatura, esfericidade, volume no eixo  $z$  e coeficientes  $k_x$  e  $k_z$ . A Tabela 9 apresenta os resultados dos algoritmos de classificação com 7 atributos.

Tabela 9 – Resultado dos classificadores com os 7 atributos selecionados com o *Extra Tree*.

Classificador	<i>F1-score</i>	Acurácia	Tempo (seg.)
K-NN (k= 1)	0,9231	92,80%	0,97
Multilayer Perceptron	0,9318	93,46%	11,91
Naive Bayes	0,9419	94,34%	0,88
<b>QDA</b>	<b>0,9649</b>	<b>96,50%</b>	<b>1,27</b>
SVM	0,9377	94,12%	1,27

O QDA apresentou o melhor resultado, com acurácia de 96,50% e *F1-score* de 0,9649. O resultado do QDA em conjunto com o seletor de atributos melhorou a exatidão do classificador em relação aos resultados apresentados na Tabela 8.

A fim de melhorar o desempenho do modelo proposto e reduzir a dimensionalidade dos atributos, o coeficiente de correlação de Pearson ( $\rho$ ) foi calculado para os 7 atributos. O objetivo foi eliminar os atributos linearmente correlacionados. Por exemplo, o coeficiente

de correlação do atributo de entropia e média da curvatura é  $\rho = -0,079$ . A reta do gráfico de regressão linear apresentada na Figura 23 mostra uma baixa correlação negativa entre esses atributos. Com a baixa correlação, os pontos no gráfico ficam dispersos, ou seja, não há relação linear entre os atributos média e entropia da curvatura, os seus valores ocorrem independentes. Além disso, o gráfico de distribuição na Figura 23 nota-se que com apenas esses dois atributos é possível distinguir várias amostras.

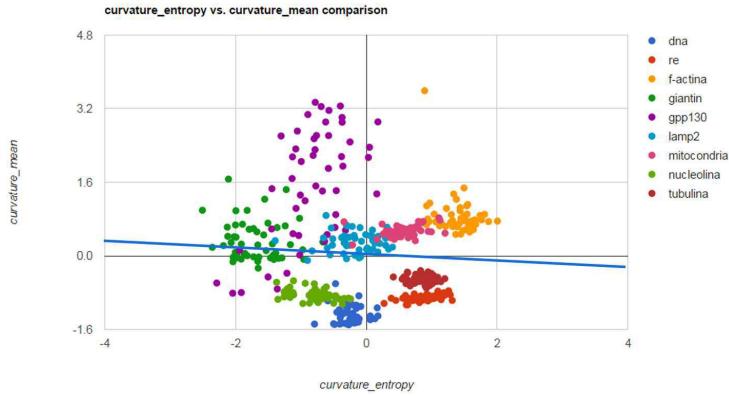


Figura 23 – Análise de regressão entre os atributos de entropia e média da curvatura.  
 $\rho = -0,079$ .

Atributos altamente correlacionados podem dificultar a classificação. Por exemplo, o coeficiente de correlação do atributos de entropia da curvatura e esfericidade é  $\rho = -0,957$ . A reta do gráfico de regressão linear apresentada na Figura 24 mostra uma alta correlação negativa entre esses atributos. A correlação negativa indica que os atributos possuem sentidos opostos, ou seja, enquanto o atributo esfericidade cresce, o atributo de entropia decresce. Além disso, a partir da distribuição na Figura 24 nota-se a dificuldade em distinguir as classes com apenas esses dois atributos.

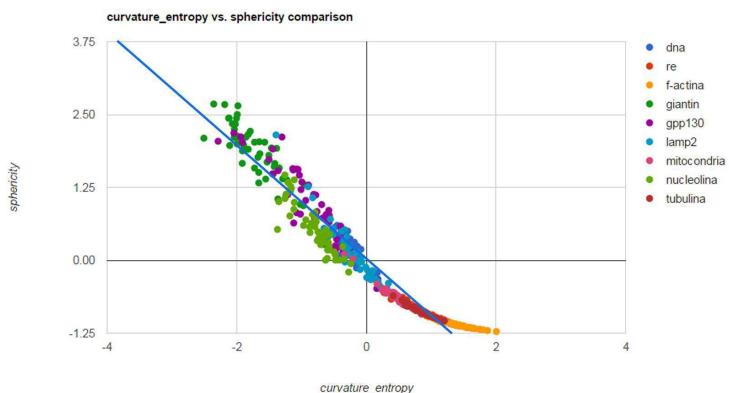


Figura 24 – Análise de regressão entre os atributos de entropia da curvatura e esfericidade.  
 $\rho = -0,957$ .

Baseando-se na correlação entre os atributos, foram selecionados manualmente 5 atributos: média, desvio padrão e entropia da curvatura, volume no eixo  $z$  e coeficiente  $k_z$ .

A Tabela 10 apresenta os resultados dos algoritmos de classificação com 5 atributos.

Tabela 10 – Resultado dos classificadores com os 5 atributos selecionados com *Extra Tree* e análise de correlação.

Classificador	<i>F1-score</i>	Acurácia	Tempo (seg.)
K-NN (k= 1)	0,9368	93,89%	0,93
Multilayer Perceptron	0,9319	93,67%	11,10
Naive Bayes	0,9460	94,78%	0,84
<b>QDA</b>	<b>0,9757</b>	<b>97,59%</b>	<b>0,92</b>
SVM	0,9392	93,31%	1,14

O QDA com os 5 atributos selecionados com árvore de decisão e análise estatística apresentou o melhor resultado com acurácia de 97,59% e *F1-score* de 0,9757. O número de atributos foi reduzido de 17 para 5, projetando um modelo de características que melhor discriminam os padrões subcelulares. Nesse modelo, percebe-se que os atributos de curvatura fornecem uma capacidade discriminatória relevante para esse problema.

A Tabela 11 apresenta a matriz de confusão resultante da validação cruzada com 10 *folds* utilizando o QDA com 5 atributos.

Tabela 11 – Matriz de confusão.

	DNA	RE	FAC	GIA	GPP	LAM	MIT	NUC	TUB
DNA	52	0	0	0	0	0	0	0	0
RE	1	49	0	0	0	0	0	0	0
FAC	0	0	49	1	0	0	0	0	0
GIA	0	0	0	49	1	0	0	0	0
GPP	0	0	0	2	46	2	0	0	0
LAM	0	0	0	0	3	47	0	0	0
MIT	0	0	0	1	0	0	49	0	0
NUC	0	0	0	0	0	0	0	50	0
TUB	0	0	0	0	0	0	0	0	52

Existe uma dificuldade em distinguir as proteínas de *giantin* e *gpp130* em uma inspeção visual. O trabalho proposto separou as duas proteínas, apenas uma amostra de *giantin* foi classificada como *gpp130* e duas amostras de *gpp130* foram classificadas como *giantin*.

De modo geral a validação cruzada realizou 454 previsões, sendo 443 corretas e apenas 11 incorretas.

A Tabela 12 apresenta os resultados de quatro abordagens de classificação de padrões subcelulares utilizando a base de dados 3D HeLa.

Tabela 12 – Comparaçāo dos resultados com outros trabalhos relacionados

Trabalho	Velliste e Murphy (2002)	Huang e Murphy (2004b)	Huang e Murphy (2004a)	Chen e Murphy (2004)	Metodologia Proposta
Algoritmo de classificação	BPNN	Majority Voting	SVM	BPNN	QDA
Qtd. atributos	28	9	17	7	5
Qtd. atributos morfológicos	5	5	5	1	5
Utiliza o DNA como referência	Sim	Sim	Não	Não	Não
Objeto de extração das medidas	Voxels do Volume 3D	Voxels do Volume 3D	Voxels do Volume 3D	Voxels do Volume 3D	Isosuperfície
Acurácia	91%	96%	89%	98%	97,59%
F1-score	não informado	não informado	não informado	não informado	0,9757
Tempo de extração dos atributos por volume (seg.)	não informado	32,0	31,5	não informado	16,01
Tempo do classificador (seg.)	não informado	1,1315	1,7708	não informado	0,92

Todas as abordagens extraíram os atributos dos voxels dos volumes 3D. Os trabalhos de Velliste e Murphy (2002) e Huang e Murphy (2004b) obtiveram o resultado utilizando a referência do DNA para a geração de atributos baseados em distância. Huang e Murphy (2004a), não utilizaram o DNA como referência para gerar atributos e obtiveram um resultado inferior. Chen e Murphy (2004), obtiveram os melhores resultados ao incluir atributos de textura de Halarick ao conjunto de atributos.

Os autores Huang e Murphy (2004b) e Huang e Murphy (2004a) utilizaram para realizar o teste de desempenho um computador com CPU de 1.7 GHz e os algoritmos foram executados com o Matlab 6.5. Neste estudo de caso utilizou-se um computador com CPU de 2.13 GHz e os algoritmos foram executados em Python. A metodologia proposta obteve um tempo médio de 16,01 segundos para gerar a isosuperfície e extrair os 17 atributos, tempo inferior aos trabalhos citados na Tabela 12. Porém, não se pode

afirmar que o algoritmo possui um desempenho melhor, pois a linguagem de programação e os recursos computacionais utilizados são diferentes.

O estudo de caso obteve resultados significativos baseados em citomorfometria, sem utilizar o DNA como referência para gerar atributos. Esse estudo de caso apresentou a média ponderada entre a precisão e a sensibilidade do classificador (*F-Score*), uma vez que um algoritmo de aprendizagem deve maximizar a precisão e a sensibilidade simultaneamente. Para comparação, foi utilizada a acurácia (mesma medida aplicada nos trabalhos relacionados).

## 5.4 Conclusão

Este estudo de caso propõe um metodologia para localização de proteínas subcelulares utilizando a citomorfometria. A extração de características em uma isosuperfície 3D pode apresentar padrões que muitas vezes não vistos em um modelo 2D ou em um volume 3D (utilização dos voxels para obter os atributos). Os atributos de curvatura, por exemplo, demonstraram ter uma alta capacidade discriminatória dos padrões subcelulares.

A redução da dimensionalidade permitiu a definição de um conjunto reduzido de atributos que melhor discriminam as classes do modelo. Utilizando um seletor híbrido com árvore de decisão e análise estatística foi possível alcançar o resultado de 97,59% de acurácia e *F1-score* de 0,9757 com o classificador QDA. O resultado obtido demonstra a capacidade dessa metodologia em reconhecer os padrões subcelulares e predizer ao biólogo a classificação de uma nova amostra.

Na metodologia proposta não é necessária o conhecimento prévio do DNA para o cálculo de novos atributos. Algumas bases de dados podem não conter o DNA. Portanto, acredita-se que ela pode ser utilizada em outros problemas, tais como, análise de deformação celular em tumores.

O resultado obtido no trabalho proposto foi ligeiramente inferior apenas ao trabalho proposto por Chen e Murphy (2004), porém, esse trabalho utilizou diversos tipos de atributos, não apenas morfológicos. Para trabalhos futuros, pretende-se realizar a análise citomorfométrica em outras bases de dados, bem como implementar outras medidas (características visuais) para fornecer mais informações aos classificadores para o aprendizado dos padrões celulares.



# CAPÍTULO 6

## **Uso da citomorfometria para a análise das alterações no padrão de heterocromatina nos cérebros de tartarugas da Amazônia**

Nessa seção será apresentado um estudo de caso em que o BioWebVis foi utilizado em todo o *pipeline* da análise citomorfométrica das alterações no padrão de heterocromatina nos cérebros de tartarugas da Amazônia.

### **6.1 Introdução**

Atualmente, a agropecuária é uma das principais atividades econômicas no Brasil e a utilização de tecnologias permite melhorias na produção de alimentos. Essas melhorias estão intimamente relacionadas com o uso de compostos químicos, tais, como herbicidas. Os herbicidas são utilizados para o controle de ervas daninhas, porém ele pode afetar organismos vivos que não são alvos, com por exemplo, organismos aquáticos (MENDONCA et al., 2016).

Os herbicidas podem alterar os processos celulares e interagir diretamente ou indiretamente com o DNA, causando instabilidades genéticas. Estudos apresentaram que animais expostos a dosagens de glifosato produziram maior lesão no DNA quando comparadas ao controle (GONZÁLEZ et al., 2013). O glifosato, de nome químico N-(fosfonometil) glicina e lançado como produto comercial de nome Roundup® é um dos herbicidas mais utilizados no Brasil. Os glifosatos utilizados na agropecuária são encontrados no recursos hídricos de diversos rios brasileiros, contaminando diversos seres vivos, tais como a *Poeclnemis expansa*, conhecida popularmente como tartaruga da Amazônia (MENDONCA et al., 2016).

A apoptose ou morte celular programada envolve a eliminação de determinadas células geneticamente. Existem várias metodologias para identificar a apoptose, tais como, alterações citomorfológicas, fragmentação do DNA, ensaio mitocondrial entre outras (ELMORE, 2007).

A metodologia baseada nas alterações citomorfológicas permite avaliar a apoptose pela detecção de eventos morfológicos. Várias alterações morfológicas ocorrem durante o processo de apoptose, tais como, o encolhimento celular, condensação da cromatina e a separação das células vizinhas (as células perderão as adesões célula-a-célula) (ELMORE, 2007).

O presente estudo de caso analisa se dosagens muito altas podem desencadear a apoptose. Foram analisadas as alterações no padrão de heterocromatina (cromatina condensada) nos cérebros de tartarugas da Amazônia. Foi utilizada uma base de dados de células extraídas dos cérebros de embriões de tartarugas expostas em diferentes concentrações de glifosato. Para analisar as alterações morfológicas, foram extraídas características utilizando a citomorfometria nas isosuperfícies das heterocromatinas. O algoritmo de agrupamento *K-Means* foi utilizado para aprender os padrões celulares e fornecer a capacidade de distingui-los. Ao final, foi calculada a *V-measure* para validar a metodologia proposta.

## 6.2 Material e Métodos

Nesta seção serão descritos os materiais e métodos utilizados para o desenvolvimento e obtenção dos resultados.

### 6.2.1 Base de Dados

Para avaliar a metodologia proposta, foi utilizada uma base de imagens 3D de células de cérebro de tartarugas da Amazônia (*Podocnemis expansa*) submetidas a diferentes dosagens de glifosato. A base de dados é composta por 8 imagens 3D divididas em 4 classes (2 imagens por classe). Cada volume 3D possui entre 25 a 100 fatias 2D e contém várias células, a dimensão dos voxels nas coordenadas *x*, *y* e *z* foram 0,379 x 0,379 x 0,988  $\mu\text{m}$ . Os substratos utilizados para a incubação dos ovos das tartarugas foram umedecidos diariamente com água destilada pura e contaminada com glifosato nas concentrações de 65, 650 e 6500 partes por bilhão (ppb), obtendo-se um grupo de controle e 3 grupos experimentais (MENDONCA et al., 2016).

A Figura 25 apresenta, para cada classe, uma imagem (uma fatia do volume 3D).

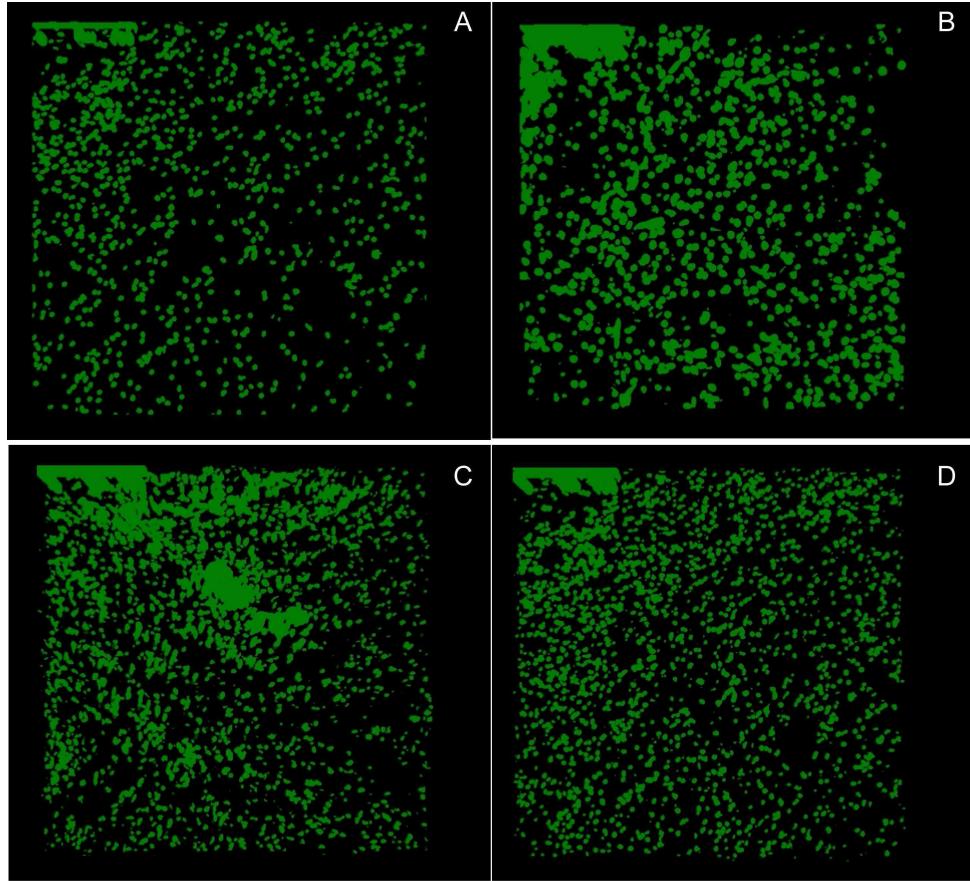


Figura 25 – Visualização da base de dados após a segmentação utilizando o software BioWebVis. (A) controle. (B) com dosagem de 65. (C) com dosagem de 650. (D) com dosagem de 6500.

As amostras estão divididas em 4 classes, cada classe representa a dosagem de herbicida submetida na amostra. As classes são: controle, com dosagem de 65, com dosagem de 650 (10x) e com dosagem de 6500 (100x).

### 6.2.2 Metodologia Proposta

As imagens foram analisadas utilizando o ambiente BioWebVis. Vários algoritmos e técnicas disponíveis no ambiente foram utilizados (i) processamento de imagens - pré-processamento, segmentação, extração de características; (ii) visualização científica - DVR, reconstrução de isosuperfície; e (iii) reconhecimento de padrões - *K-means*. O *pipeline* geral para análise dos dados é mostrado na Figura 26.

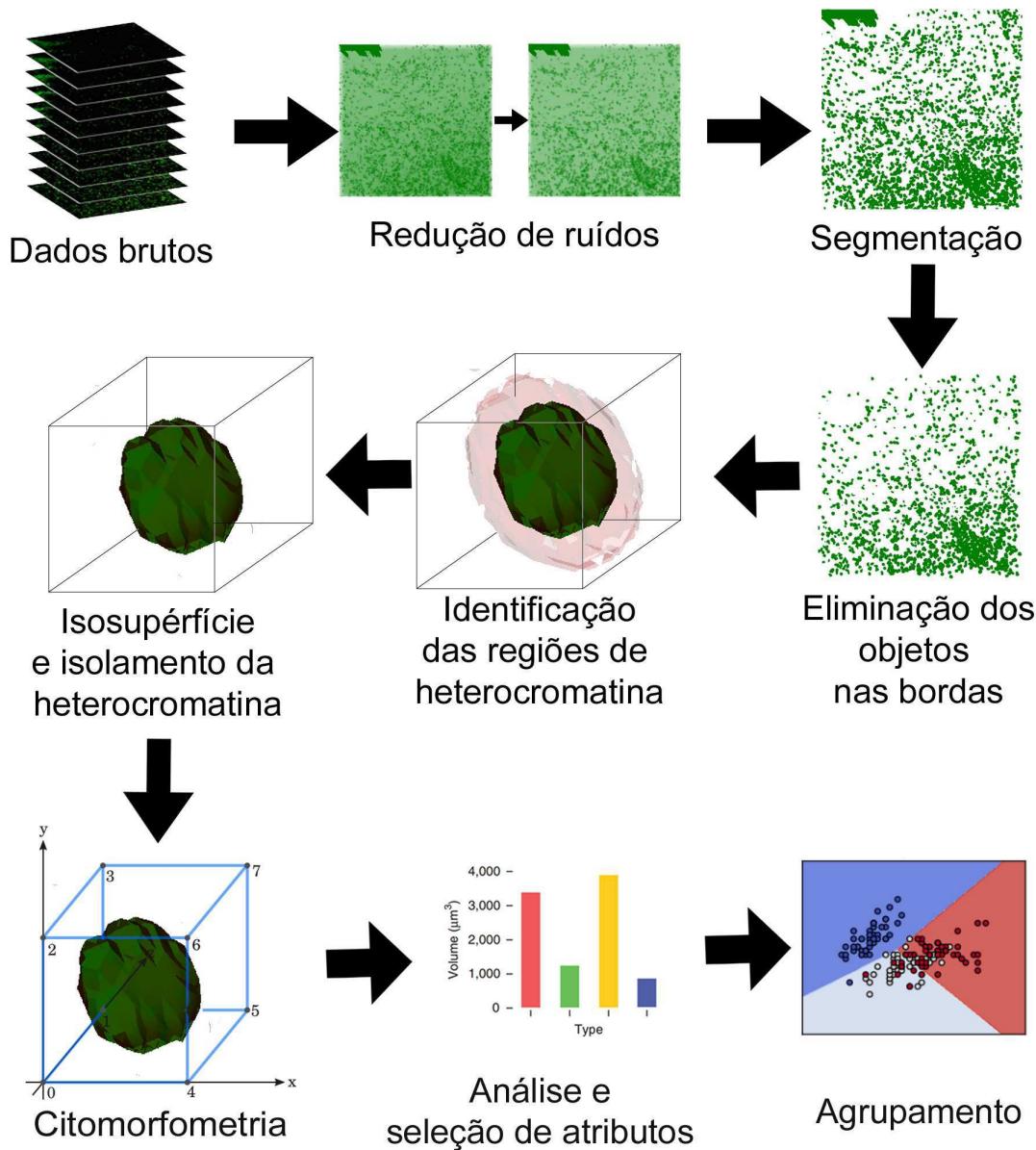


Figura 26 – Pipeline geral utilizado para análise das alterações no padrão de heterocromatina nos cérebros de tartarugas.

Primeiramente, foi realizado a redução de ruídos nos dados brutos (volume 3D) utilizando o filtro Gaussiano. Na redução de ruídos estabeleceu-se o *kernel* igual a  $\sigma = 2$ . Esse parâmetro foi definido com base nos testes realizados, e com  $\sigma = 2$  obteve-se melhores resultados. Em seguida, a imagem foi segmentada utilizando o limiar igual a 10, determinado com base em testes realizados. Inicialmente, o limiar foi obtido automaticamente com o método de Otsu, porém o tamanho das células ficaram bem reduzidas (Figura 27 (D)), dificultando identificação das regiões de heterocromatina.

A Figura 27 (C) apresenta a imagem final após a redução de ruídos e segmentada com o limiar igual a 10.

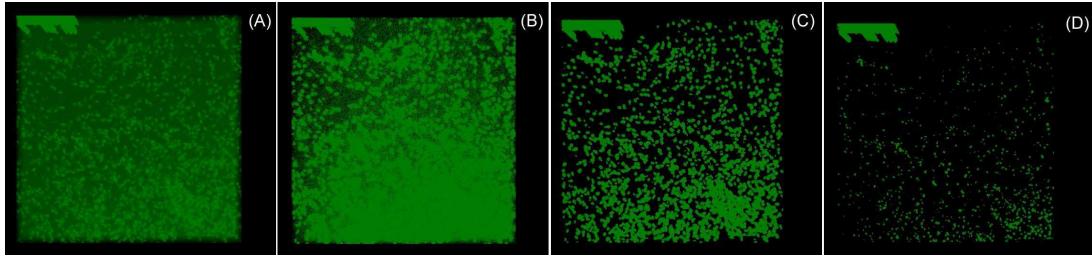


Figura 27 – Visualização da amostra de células de tartaruga após a segmentação utilizando o software BioWebVis. (A) Imagem original. (B) Imagem sem redução de ruídos e segmentada com limiar igual a 10. (C) Imagem com a redução de ruídos e segmentada com limiar igual a 10. (D) Imagem com a redução de ruídos e segmentada com o método de Otsu.

Após a segmentação, foram eliminadas as células incompletas ou objetos irrelevantes. Para isso, foram descartados os objetos que estavam na borda do volume, ou seja, os objetos em que algum voxel esteja na posição inicial ou final dos eixos  $x$ ,  $y$  e  $z$ . No processo de eliminação foi realizada a separação dos objetos e para cada objeto foi verificado se algum voxel faz parte das extremidades do volume. Se o objeto possui algum voxel nas extremidades, então esse objeto é descartado do volume.

O próximo passo foi identificar as regiões de heterocromatina. A heterocromatina é uma região densamente colorada dentro da célula, para identificá-la foi necessário encontrar as regiões com intensidades mais escuras dentro da célula no volume 3D. Foram realizados os seguintes passos para encontrar as regiões de heterocromatina:

1. Realizar a separação das células no volume 3D;
2. Para cada célula obter o valor médio da intensidade do voxels (escala de cinza);
3. Recuperar as células com as 5 maiores médias das intensidades;
4. Realizar uma nova média com as intensidades das 5 células selecionadas no item 3;
5. Aplicar a segmentação no volume 3D com o limiar obtido pela média no item 4.

Após encontrar as regiões de heterocromatina, foi realizado a reconstrução da isosuperfície da amostra. A Figura 28 apresenta a isosuperfície de uma célula, em que é possível visualizar a região da heterocromatina (destacada na isosuperfície de cor verde) e o restante da região celular (destacada na isosuperfície de cor avermelhada).

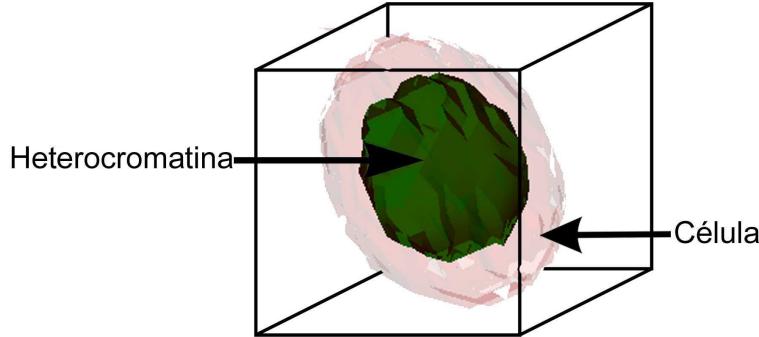


Figura 28 – Visualização heterocromatina de uma célula utilizando o software BioWebVis.

Para cada região de heterocromatina, 17 atributos foram extraídos: área da superfície, volume, volume no eixo  $x$ , volume no eixo  $y$  e volume no eixo  $z$ , curvatura (média, mediana, desvio padrão, entropia), NSI, esfericidade, convexidade (por área da superfície e por volume), coeficientes  $k_x$ ,  $k_y$ , e  $k_z$  e diâmetro esférico equivalente.

As regiões em que a área da superfície não estavam entre 100 e 300 foram eliminadas. Esse processo foi realizado pois as amostras possuem várias regiões grudadas que influenciam no processo de agrupamento por ter valores anormais. Esse limiar foi definido com base nos testes realizados. Cada amostra possui diversas células, o valor médio foi calculado para cada atributo da amostra. Os atributos extraídos de cada amostra estão em escalas diferentes. O  $Z$ -Score disponível no BioWebVis foi utilizado no processo de normalização para criar uma escala comum.

Após a normalização, os dados compartilham uma escala comum, que é importante para o reconhecimento de padrões. O próximo passo consiste em realizar a seleção dos atributos, processo para encontrar o conjunto de características que melhor separe as classes. Foi utilizado o PCA disponível no BioWebVis para encontrar o melhor conjunto de características que representam cada classe.

Em seguida, o algoritmo de agrupamento  $K$ -Means disponível no BioWebVis foi utilizado para agrupar as amostras e encontrar padrões nos dados. Foi utilizado a distância Euclidiana como medida de similaridade no  $K$ -Means. Para avaliar o desempenho do algoritmo foi utilizado o  $V$ -measure.

## 6.3 Resultados

Com os 17 atributos das 8 amostras extraídos e normalizados, foi realizado o agrupamento utilizando o  $K$ -Means. Em todos os testes, foram informados os seguintes parâmetros ao algoritmo: número de grupos igual a 4 e número máximo de interações igual a 100. O número de grupos indica ao algoritmo para separar as amostras em 4 grupos, sendo eles: controle, com dosagem de 65, 650 e 6500.

A Tabela 13 apresenta os resultados do agrupamento com o algoritmo  $K$ -Means.

Tabela 13 – Resultado do agrupamento.

Qtd. de atributos	<i>V-measure</i>	Tempo (seg.)
17	0,84	0,45
1 (obtido com PCA)	1,0	0,11
1 (área da superfície)	1,0	0,11

A medida *V-measure* varia de 0 à 1, em que 1 representa um agrupamento perfeito. Para melhorar o desempenho do algoritmo de agrupamento e reduzir a dimensionalidade foi utilizado o PCA. Com 1 atributo obtido com o PCA e com apenas o atributo área da superfície obteve-se o resultado igual a 1, ou seja, o algoritmo conseguiu agrupar todas as amostras de classes iguais no mesmo grupo.

A metodologia proposta obteve o tempo médio por amostra de 318 segundos (5 minutos e 20 segundos) para realizar os processos do *pipeline* até a extração dos 17 atributos. Cada amostra possui diversas células, para cada célula foi necessária a análise citomorfométrica, consumindo a maior parte do tempo do processamento, visto que na base de dados possui amostras com mais de 400 células. Nesse estudo de caso utilizou-se um computador com CPU de 2.13 GHz e os algoritmos foram executados em Python.

Ao utilizar o PCA para gerar novos atributos com o objetivo de diminuir a dimensionalidade, observou-se que gerando apenas um novo componente obteve-se resultados satisfatórios e que provavelmente existe um ou mais atributos que tenham uma alta capacidade de discriminação no modelo de dados. Para identificar esses atributos foi utilizado o gráfico de dispersão.

Ao analisar o gráfico de dispersão dos atributos, foi identificado que o volume e a área da superfície possuem alta capacidade de distinguir as amostras. Com apenas um desses dois atributos é possível realizar o agrupamento das amostras com o *V-measure* igual a 1.

Esses dois atributos são altamente correlacionados, o coeficiente de correlação deles é  $\rho = 0,99$ . A reta do gráfico de regressão linear apresentada na Figura 29 mostra uma alta correlação positiva entre esses atributos. A correlação positiva indica que os atributos estão no mesmo sentido, ou seja, enquanto o atributo volume cresce, o atributo de área da superfície também cresce.

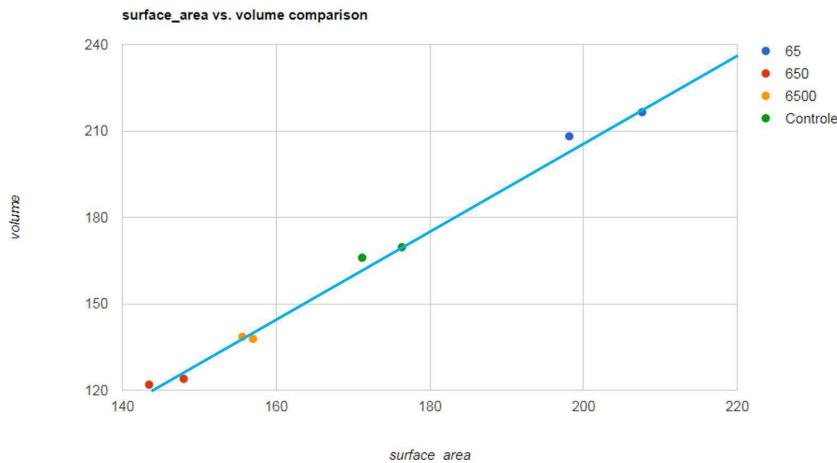


Figura 29 – Análise de regressão entre os atributos da área da superfície e volume.  $\rho = 0,99$

Além disso, a partir da distribuição na Figura 29 nota-se que em relação ao controle o volume e a área da superfície diminuiriam nas amostras com dosagens altas, e com dosagens baixas eles aumentaram. Porém, isso não poderá comprovar se houve a apoptose, pois será necessário a análise de um biólogo e também provavelmente a utilização de mais metodologias para essa comprovação. A interpretação biológica desses resultados ainda está sob investigação.

## 6.4 Conclusão

Este estudo de caso propõe um metodologia para analisar as alterações morfológicas na heterocromatina. A análise citomorfométrica nas isosuperfícies 3D permitiu identificar que nas dosagens altas de herbicidas ocorreu o encolhimento celular, uma característica presente no processo de apoptose. Porém, para a comprovação da apoptose é necessária interpretação biológica e provavelmente a utilização de outras metodologias.

A redução da dimensionalidade permitiu a definição de um conjunto reduzido de atributos que melhor discriminam as classes do modelo. Utilizando PCA foi obtido um único componente, resultando no *V-measure* igual a 1. Os atributos de área da superfície e volume foram dominantes no conjunto de atributos extraídos, consequentemente eles estão diretamente ligados com o processo de encolhimento celular. O resultado obtido demonstra a capacidade dessa metodologia em reconhecer as alterações morfológicas na heterocromatina.

Com a dosagem de 65 (ppb), houve um aumento na área da superfície e no volume em relação ao controle. Será necessário a análise de mais amostras de células de controle e de células dosagem com de 65 (ppb), pois dependendo das amostras analisadas poderá ser concluído que em dosagens baixas não há alterações morfológicas na heterocromatina.

Para trabalhos futuros, pretende-se analisar um conjunto maior de amostras de células de cérebros de tartarugas, utilizando algoritmos de classificação supervisionada e também confrontar os resultados obtidos com a interpretação biológica.



# CAPÍTULO 7

## Conclusão

A maioria dos softwares de bioimagens são disponibilizados como aplicação local. Com a evolução das tecnologias da internet foi possível desenvolver um software de bioimagem na nuvem com características presentes nesses softwares locais, e ainda acrescentar recursos que não estão presentes nos trabalhos relacionados disponibilizados na nuvem, tal como, a renderização de grandes conjuntos de dados.

A usabilidade foi um fator relevante ao propor o ambiente, pois a disponibilização dos dados na nuvem contribuirá com a usabilidade do software proposto, evitando a instalação e configuração. Além disso, o ambiente fornece um repositório para o usuário armazenar seus resultados e amostras. O ambiente suporta vários formatos, pois fornecedores de microscópicos disponibilizam as imagens digitalizadas em formatos abertos, porém outras já disponibilizam em formatos vinculados com o microscópico. A restrição de formatos pode ser um fator que limite o uso do ambiente proposto.

A limitação do ambiente proposto está relacionada com a velocidade da internet, principalmente na tarefa de *upload*, pois o envio de grandes conjuntos de dados poderá consumir um bom tempo dependendo da velocidade da internet. Os requisitos mínimos do servidor do BioWebVis dependerá da quantidade de usuários, pois todo processamento é realizado no servidor. Por outro lado a arquitetura na nuvem irá permitir a escalabilidade, ou seja, o aumento do poder computacional de forma simples.

O ambiente proposto é capaz de fornecer informações relevantes ao usuário na tomada de decisões, porém ele não substitui um especialista. O ambiente foi capaz de realizar dois experimentos distintos e com metodologias diferentes, fornecendo resultados satisfatórios. Os estudos de casos permitiram validar o ambiente e propor metodologias baseadas na extração de atributos de citomorfometria em imagens 3D. A utilização das imagens 3D permitiu a visualização de comportamentos celulares que provavelmente não seriam vistos em imagens 2D.

## 7.1 Principais Contribuições

Dentre as principais contribuições deste trabalho pode-se citar as metodologias propostas nos dois estudos de casos. As metodologias desenvolvidas produziram resultados satisfatórios, que podem auxiliar os especialistas na tomada de decisões. Elas ainda contribuiriam para validar o ambiente e gerar as necessidades de desenvolvimento das funcionalidades do BioWebVis. O trabalho também contribui com um software de bioimagem na nuvem.

## 7.2 Trabalhos Futuros

Esse trabalho propôs um ambiente na nuvem para contribuir com a usabilidade, evitando problemas de instalações. Porém, nesse trabalho não foi realizado testes de avaliação de usabilidade com os usuários. Como trabalho futuro, pretende-se realizar testes de usabilidade em campo com os usuários finais para validar o ambiente e compreender melhor a interação entre pessoas e computadores. Baseando-se no *feedback* fornecido pela avaliação da usabilidade, pretende-se realizar um estudo sobre Interação humano-computador (IHC), com o objetivo de aplicar técnicas para redesenhar as telas, minimizando possíveis dificuldades na utilização do BioWebVis.

Pretende-se também tornar o ambiente flexível, permitindo que os desenvolvedores adicionem novos algoritmos por meio de *plug-ins*. Assim, eles poderão contribuir com o ambiente, enriquecendo a gama de algoritmos disponíveis. Também será desenvolvido a visualização dos volumes sobre o tempo, fornecendo mais informações para o cientistas em suas análises.

O servidor do BioWebVis deverá ser compilado para as plataformas Linux e iOS, visto que organizações podem querer disponibilizar o ambiente em suas redes internas e limitar a plataforma apenas para Windows pode influenciar na escolha do ambiente.

## 7.3 Contribuições em Produção Bibliográfica

Como produção bibliográfica resultante deste trabalho tem-se um artigo publicado e um submetido:

- ❑ SILVA, E. H.; BELETTI, M. E.; SOUZA, J. R.; TRAVENCOLO, B. A. N. Ambiente web para citomorfometria usando imagens 3D. *Journal of Health Informatics*, v. 8, p. 643, Suplemento I - XV Congresso Brasileiro de Informática em Saúde - CBIS 2016.

- SILVA, E. H.; SOUZA, J. R.; TRAVENCOLO, B. A. N. Use of cytomorphometry for classification of subcellular patterns in 3D images. Journal Applied Artificial Intelligence (**Submetido**).



---

## Referências

- ALLAN, C. et al. Omero: flexible, model-driven data management for experimental biology. **Nature methods**, Nature Publishing Group, v. 9, n. 3, p. 245–253, 2012.
- ALYASSIN, A. M. et al. Evaluation of new algorithms for the interactive measurement of surface area and volume. **Medical physics**, American Association of Physicists in Medicine, v. 21, n. 6, p. 741–752, 1994.
- BACUS, J. W. et al. Image morphometric nuclear grading of intraepithelial neoplastic lesions with applications to cancer chemoprevention trials. **Cancer Epidemiology Biomarkers & Prevention**, AACR, v. 8, n. 12, p. 1087–1094, 1999.
- BARBOSA, E. B. et al. Proteômica: metodologias e aplicações no estudo de doenças humanas. **Revista da Associação Médica Brasileira**, Elsevier BV, v. 58, n. 3, p. 366–375, may 2012.
- BEHR, J. et al. X3dom: a dom-based html5/x3d integration model. In: **ACM Proceedings of the 14th International Conference on 3D Web Technology**. [S.l.], 2009. p. 127–135.
- BILGIN, C. C. et al. Biosig3d: High content screening of three-dimensional cell culture models. **PloS one**, Public Library of Science, v. 11, n. 3, p. e0148379, 2016.
- BISHOP, C. **Pattern Recognition and Machine Learning**. [S.l.]: Springer, 2006. (Information Science and Statistics). ISBN 9780387310732.
- \_\_\_\_\_. **Pattern Recognition and Machine Learning (Information Science and Statistics)**, 1st edn. 2006. corr. 2nd printing edn. [S.l.]: Springer, New York, 2007.
- BOSMA, M. K.; SMIT, J.; LOBREGT, S. Iso-surface volume rendering. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. **Medical Imaging'98**. [S.l.], 1998. p. 10–19.
- CABELLO, R. Three.js. URL: <https://github.com/mrdoob/three.js>, 2010.
- CARDONA, A.; TOMANCAK, P. Current challenges in open-source bioimage informatics. **Nature methods**, Nature Publishing Group, v. 9, n. 7, p. 661–665, 2012.
- CHAUMONT, F. D. et al. Icy: an open bioimage informatics platform for extended reproducible research. **Nature methods**, Nature Publishing Group, v. 9, n. 7, p. 690–696, 2012.

- CHEN, X.; MURPHY, R. Robust classification of subcellular location patterns in high resolution 3d fluorescence microscope images. In: **The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society**. [S.l.]: Institute of Electrical and Electronics Engineers (IEEE), 2004.
- CHITYALA, R.; PUDIPEDDI, S. **Image Processing and Acquisition using Python**. [S.l.]: CRC Press, 2015. (Chapman & Hall/CRC Mathematical and Computational Imaging Sciences Series). ISBN 9781498760577.
- COLLINS, T. J. et al. Imagej for microscopy. **Biotechniques**, v. 43, n. 1 Suppl, p. 25–30, 2007.
- CONGOTE, J. Medx3dom: Medx3d for x3dom. In: ACM. **Proceedings of the 17th International Conference on 3D Web Technology**. [S.l.], 2012. p. 179–179.
- COSTA, L. d. F.; JUNIOR, R. M. C. **Shape analysis and classification: theory and practice**. [S.l.]: CRC press, 2010.
- CUNNINGHAM, P.; DELANY, S. J. k-nearest neighbour classifiers. **Multiple Classifier Systems**, Springer, p. 1–17, 2007.
- ELICEIRI, K. W. et al. Biological imaging software tools. **Nature methods**, Nature Publishing Group, v. 9, n. 7, p. 697–710, 2012.
- ELMORE, S. Apoptosis: a review of programmed cell death. **Toxicologic pathology**, Sage Publications, v. 35, n. 4, p. 495–516, 2007.
- ELVINS, T. T. A survey of algorithms for volume visualization. **ACM Siggraph Computer Graphics**, ACM, v. 26, n. 3, p. 194–201, 1992.
- FETTE, I.; MELNIKOV, A. The websocket protocol. 2011.
- FISHER, J. Visualizing the connection among convex hull, voronoi diagram and delaunay triangulation. In: **37th Midwest Instruction and Computing Symposium**. [S.l.]: s.n., 2004.
- GALLAGHER, R.; PRESS, S. **Computer Visualization: Graphics Techniques for Engineering and Scientific Analysis**. [S.l.]: Taylor & Francis, 1994. ISBN 9780849390500.
- GEURTS, P.; ERNST, D.; WEHENKEL, L. Extremely randomized trees. **Machine Learning**, Springer Nature, v. 63, n. 1, p. 3–42, mar 2006.
- GONZÁLEZ, E. L. et al. Induction of micronuclei in broad snouted caiman (caiman latirostris) hatchlings exposed in vivo to roundup®(glyphosate) concentrations used in agriculture. **Pesticide biochemistry and physiology**, Elsevier, v. 105, n. 2, p. 131–134, 2013.
- GONZALEZ, R.; WOODS, R. **Processamento Digital de Imagens**. [S.l.]: Pearson, 2010. ISBN 9788576054016.
- GUILHERMANO, L. et al. **Páginas da história da medicina**. [S.l.]: EDIPUCRS, 2010. ISBN 9788574309743.

- HABER, R. B.; MCNABB, D. A. Visualization idioms: A conceptual model for scientific visualization systems. **Visualization in scientific computing**, v. 74, p. 93, 1990.
- HAEHN, D. et al. Neuroimaging in the browser using the x toolkit. **Frontiers in Neuroinformatics**, v. 101, 2014.
- HALL, M. A. **Correlation-based feature selection for machine learning**. Tese (Doutorado) — The University of Waikato, 1999.
- HEARST, M. A. et al. Support vector machines. **IEEE Intelligent Systems and their Applications**, IEEE, v. 13, n. 4, p. 18–28, 1998.
- HUANG, K.; MURPHY, R. F. Boosting accuracy of automated classification of fluorescence microscope images for location proteomics. **BMC Bioinformatics**, Springer Nature, v. 5, n. 1, p. 78, 2004.
- \_\_\_\_\_. From quantitative microscopy to automated image understanding. **Journal of Biomedical Optics**, SPIE-Intl Soc Optical Eng, v. 9, n. 5, p. 893, 2004.
- JACINTO, H. et al. A web interface for 3d visualization and interactive segmentation of medical images. In: ACM. **Proceedings of the 17th International Conference on 3D Web Technology**. [S.l.], 2012. p. 51–58.
- JENI, L. A.; COHN, J. F.; TORRE, F. D. L. Facing imbalanced data—recommendations for the use of performance metrics. In: **2013 Humaine Association Conference on Affective Computing and Intelligent Interaction**. [S.l.]: Institute of Electrical and Electronics Engineers (IEEE), 2013.
- JOURDAIN, S.; AYACHIT, U.; GEVECI, B. Paraviewweb: A web framework for 3d visualization and data processing. In: **IADIS international conference on web virtual reality and three-dimensional worlds**. [S.l.: s.n.], 2010. v. 7, p. 1.
- JUNG, Y. et al. Declarative 3d approaches for distributed web-based scientific visualization services. In: **Dec3D**. [S.l.: s.n.], 2012.
- KANKAANPÄÄ, P. et al. Bioimagexd: an open, general-purpose and high-throughput image-processing platform. **Nature methods**, Nature Publishing Group, v. 9, n. 7, p. 683–689, 2012.
- KASSIMI, M. A. et al. 3d model classification and retrieval based on semantic and ontology. Citeseer, 2011.
- KVILEKVAL, K. et al. Bisque: a platform for bioimage analysis and management. **Bioinformatics**, Oxford Univ Press, v. 26, n. 4, p. 544–552, 2010.
- LAMPRECHT, M. R. et al. Cellprofiler: free, versatile software for automated biological image analysis. **Biotechniques**, EATON PUBLISHING, v. 42, n. 1, p. 71, 2007.
- LIU, H.; MOTODA, H. **Computational Methods of Feature Selection**. [S.l.]: CRC Press, 2007. (Chapman & Hall/CRC Data Mining and Knowledge Discovery Series). ISBN 9781584888796.
- LORENSEN, W. E.; CLINE, H. E. Marching cubes: A high resolution 3d surface construction algorithm. In: ACM. **ACM siggraph computer graphics**. [S.l.], 1987. v. 21, n. 4, p. 163–169.

- LOYEK, C. et al. Bioimax: A web 2.0 approach for easy exploratory and collaborative access to multivariate bioimage data. **BMC bioinformatics**, BioMed Central Ltd, v. 12, n. 1, p. 297, 2011.
- MARION, C.; JOMIER, J. Real-time collaborative scientific webgl visualization with websocket. In: ACM. **Proceedings of the 17th international conference on 3D web technology**. [S.l.], 2012. p. 47–50.
- MEIJERING, E. et al. Imagining the future of bioimage analysis. **Nature Biotechnology**, Nature Research, v. 34, n. 12, p. 1250–1255, 2016.
- MENDONCA, J. dos S. et al. Effects of the exposure to atrazine and glyphosate throughout incubation on bone development of podocnemis expansa (testudines, podocnemididae). **International Journal of Pure and Applied Zoology**, Biomedical Research, 2016.
- MESMOUDI, M. M.; FLORIANI, L. D.; MAGILLO, P. Discrete curvature estimation methods for triangulated surfaces. In: **Applications of Discrete Geometry and Mathematical Morphology**. [S.l.]: Springer, 2012. p. 28–42.
- MEYER, M. et al. Discrete differential-geometry operators for triangulated 2-manifolds. In: **Visualization and mathematics III**. [S.l.]: Springer, 2003. p. 35–57.
- OTSU, N. A threshold selection method from gray-level histograms. **IEEE transactions on systems, man, and cybernetics**, IEEE, v. 9, n. 1, p. 62–66, 1979.
- PENG, H. et al. Extensible visualization and analysis for multidimensional images using vaa3d. **Nature protocols**, Nature Publishing Group, v. 9, n. 1, p. 193–208, 2014.
- PIEPER, S.; HALLE, M.; KIKINIS, R. 3d slicer. In: IEEE. **Biomedical Imaging: Nano to Macro, 2004. IEEE International Symposium on**. [S.l.], 2004. p. 632–635.
- PIROOZNIA, M. et al. A comparative study of different machine learning methods on microarray gene expression data. **BMC Genomics**, Springer Nature, v. 9, n. Suppl 1, p. S13, 2008.
- RINGNÉR, M. What is principal component analysis? **Nature biotechnology**, New York, NY: Nature Pub. Co., 1996-, v. 26, n. 3, p. 303–304, 2008.
- RISH, I. An empirical study of the naive bayes classifier. In: IBM NEW YORK. **IJCAI 2001 workshop on empirical methods in artificial intelligence**. [S.l.], 2001. v. 3, n. 22, p. 41–46.
- ROSENBERG, A.; HIRSCHBERG, J. V-measure: A conditional entropy-based external cluster evaluation measure. In: . [S.l.: s.n.], 2007. v. 7, p. 410–420.
- SCHINDELIN, J. et al. Fiji: an open-source platform for biological-image analysis. **Nature methods**, Nature Publishing Group, v. 9, n. 7, p. 676–682, 2012.
- SCHROEDER, W. J.; LORENSEN, B.; MARTIN, K. **The visualization toolkit**. [S.l.]: Kitware, 2004.

- SEABRA, A. de. **Estatística empresarial**. FGV, 2015. (Práticas de gestão). ISBN 9788522516902. Disponível em: <<https://books.google.com.br/books?id=liyHCgAAQBAJ>>.
- SRIVASTAVA, S.; GUPTA, M. R.; FRIGYIK, B. A. Bayesian quadratic discriminant analysis. **Journal of Machine Learning Research**, v. 8, n. Jun, p. 1277–1305, 2007.
- SUBRAMANIAN, G. **Python Data Science Cookbook**. Packt Publishing, 2015. ISBN 9781784393663. Disponível em: <<https://books.google.com.br/books?id=kPOoCwAAQBAJ>>.
- THEODORIDIS, S.; KOUTROUMBAS, K. **Pattern Recognition, Third Edition**. Orlando, FL, USA: Academic Press, Inc., 2006. ISBN 0123695317.
- TRUE, L. D. Morphometric applications in anatomic pathology. **Human pathology**, Elsevier, v. 27, n. 5, p. 450–467, 1996.
- VAKIL, R. **A Mathematical Mosaic: Patterns & Problem Solving**. [S.l.]: Brendan Kelly Pub., 1996. ISBN 9781895997040.
- VELLISTE, M.; MURPHY, R. Automated determination of protein subcellular locations from 3d fluorescence microscope images. In: **Proceedings IEEE International Symposium on Biomedical Imaging**. [S.l.]: Institute of Electrical and Electronics Engineers (IEEE), 2002.
- WALDEMARIN, K. C. A.; BELETTI, M. E.; COSTA, L. F. Nuclear morphometry of neoplastic cells as a method for diagnosis of histiocytoma, mastocytoma and transmissible venereal tumor in dogs. **Real-Time Imaging**, Elsevier, v. 10, n. 4, p. 197–204, 2004.
- WARE, C. **Information Visualization: Perception for Design**. [S.l.]: Elsevier Science, 2012. (Interactive Technologies). ISBN 9780123814654.
- WEIBEL, E. R. Stereological principles for morphometry. **Int rev cytol**, v. 26, p. 235–302, 1969.
- XIAOJUN, C. et al. An integrated surgical planning and virtual training system using a force feedback haptic device for dental implant surgery. In: **IEEE. Audio Language and Image Processing (ICALIP), 2010 International Conference on**. [S.l.], 2010. p. 1257–1261.
- ZUNIC, J.; ROSIN, P. L. A new convexity measure for polygons. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, IEEE, v. 26, n. 7, p. 923–934, 2004.