



UNIVERSIDADE ESTADUAL PAULISTA
“JÚLIO DE MESQUITA FILHO”
Câmpus de São José do Rio Preto

Rafael Silva Santos

Uma abordagem de exploração volumétrica baseada em
agrupamento e redução dimensional para apoiar a definição de
funções de transferência multidimensionais

São José do Rio Preto
2018

Rafael Silva Santos

Uma abordagem de exploração volumétrica baseada em
agrupamento e redução dimensional para apoiar a definição de
funções de transferência multidimensionais

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Câmpus de São José do Rio Preto.

Financiadora: CAPES

Orientador: Prof. Dr. Danilo Medeiros Eler

São José do Rio Preto
2018

Santos, Rafael Silva.

Uma abordagem de exploração volumétrica baseada em agrupamento e redução dimensional para apoiar a definição de funções de transferência multidimensionais / Rafael Silva Santos. -- São José do Rio Preto, 2018
89 f. : il., tabs.

Orientador: Danilo Medeiros Eler

Dissertação (mestrado) – Universidade Estadual Paulista “Júlio de Mesquita Filho”, Instituto de Biociências, Letras e Ciências Exatas

1. Computação gráfica. 2. Visualização da informação.
3. Processamento de imagens - Técnicas digitais. 4. Algoritmos de computador. I. Universidade Estadual Paulista "Júlio de Mesquita Filho". Instituto de Biociências, Letras e Ciências Exatas. II. Título.

CDU – 518.72:76

Ficha catalográfica elaborada pela Biblioteca do IBILCE
UNESP - Câmpus de São José do Rio Preto

Rafael Silva Santos

Uma abordagem de exploração volumétrica baseada em
agrupamento e redução dimensional para apoiar a definição de
funções de transferência multidimensionais

Dissertação apresentada como parte dos
requisitos para obtenção do título de Mestre
em Ciência da Computação, junto ao
Programa de Pós-Graduação em Ciência da
Computação, do Instituto de Biociências,
Letras e Ciências Exatas da Universidade
Estadual Paulista “Júlio de Mesquita
Filho”, Campus de São José do Rio Preto.

Financiadora: CAPES

Comissão Examinadora

Prof. Dr. Danilo Medeiros Eler
UNESP – FCT – Câmpus de Presidente Prudente
Orientador

Prof^ª. Dr^ª. Carla Maria Dal Sasso Freitas
UFRGS – INF – Câmpus do Vale

Prof. Dr. Milton Hirokazu Shimabukuro
UNESP – FCT – Câmpus de Presidente Prudente

Presidente Prudente
27 de março de 2018

Aos meus pais, Amilton e Angelina

Agradecimentos

Primeiramente, a Deus que é o Senhor de todas as coisas.

Aos meus pais, Amilton e Angelina, pelo imensurável apoio durante o desenvolvimento deste trabalho e durante toda a minha vida.

À minha namorada, Mayara, pelo apoio e paciência pela minha ausência nesses últimos anos.

Ao professor Danilo pela oportunidade, orientação e acompanhamento desde a época da graduação. É um profissional incrível e uma inspiração para quem está começando a trajetória no meio acadêmico.

Aos meus amigos por todas às vezes que tornei rigorosa minha dedicação ao trabalho e por, ainda assim, sempre me apoiarem.

À minha família, principalmente à minha madrinha, Tia Nega, por todo carinho e ajuda no que foi possível.

À UNESP pela minha sólida formação acadêmica e iniciação à docência, mesmo diante de todos os percalços que assolam as universidades públicas do Brasil. Especialmente, aos professores e funcionários do Departamento de Matemática e Computação da Faculdade de Ciências e Tecnologia. Ao professor Cássio pela minha iniciação à pesquisa. Ao professor Rogério pelas contribuições durante minha trajetória acadêmica. Ao Fernando Pacanelli pela amizade e pelo apoio durante todo o tempo que estive na universidade.

Aos amigos que fiz dentro da universidade. Principalmente, ao Pedro por me acompanhar durante as viagens a Bauru e São José do Rio Preto. Além disso, ao Wilson pelo companheirismo de laboratório e pelas pontuais contribuições neste trabalho.

À CAPES pelo apoio financeiro.

A todos que de alguma forma contribuíram para que fosse possível chegar até aqui.

*“Só podemos ver uma curta distância à frente, mas o suficiente para perceber que há
muito a se fazer.” – Alan Turing*

Resumo

Funções de transferência (FTs) são uma parte crucial do processo de exploração volumétrica em Visualização Direta de Volumes. Nesse processo, FTs desempenham duas tarefas principais: a classificação de materiais e o mapeamento de informações presentes nos dados para propriedades visuais. A busca por uma solução que lide com ambas as tarefas envolve uma série de fatores que, em conjunto, são um dos maiores desafios de visualização volumétrica. Neste trabalho, propomos uma abordagem de exploração que tem por objetivo envolver todo escopo e simplificar tanto a definição de FTs multidimensionais quanto a manipulação de *datasets*. A abordagem se organiza em três componentes: uma heurística baseada em entropia e correlação que guia a seleção de atributos para formação do espaço de entrada; um método de classificação que emprega a técnica de redução de dimensionalidade *FastMap* e a técnica de agrupamento DBSCAN para proporcionar a descoberta semiautomática de características volumétricas; e uma interface simplificada que, atrelada ao método de classificação, produz um gráfico de dispersão 2D de características para a exploração do volume. Inicialmente, o usuário deve analisar o *ranking* de atributos para formar um espaço multidimensional. Depois, deve escolher parâmetros para gerar o gráfico de características. Finalmente, deve navegar por esse gráfico a fim de identificar materiais ou estruturas relevantes. Nos experimentos realizados para avaliar a abordagem, os mecanismos disponibilizados permitiram encontrar e isolar de forma efetiva características inseridas em todos os *datasets* investigados. Aponta-se ainda como contribuição o baixo custo computacional, na prática, a complexidade de tempo do método de classificação é de $O(n \log n)$. O tempo de execução foi inferior a 11 segundos, mesmo quando *datasets* formados por cerca de 10 milhões de instâncias e com mais de 10 dimensões são utilizados.

Palavras-chave: função de transferência. especificação de funções de transferência. agrupamento. redução dimensional.

Abstract

Transfer functions (TFs) are a crucial part of the volume exploration process in Direct Volume Rendering. In this process, TFs perform two main tasks: material classification and mapping of information to visual properties. The search for a solution that copes with both tasks involves a number of factors that, together, is one of the greatest challenges of volume visualization. In this work, we propose an exploration approach that aims to involve the entire scope and a simplify both the definition of multidimensional TFs and the manipulation of datasets. The approach is organized into three components: a heuristic based on entropy and correlation that guides the selection of attributes to conceive the input space; a classification method, which uses the dimensionality reduction technique FastMap and the clustering technique DBSCAN to provide a semiautomatic features finding; and a simplified interface that, linked to the previous method, provide a 2D scatter plot of features for volume exploration. Initially, the user must analyze at the ranking of attributes to form a multidimensional space. Afterwards, it must choice parameters to generate the scatter plot. Finally, it must navigate through this chart in order to reveal relevant materials and features. In the experiments performed to evaluate the approach, the available mechanisms allow to effectively find and isolate features inserted in all investigated datasets. It is also pointed out as contribution a low computational cost, in practice the time complexity of the classification method is $O(n \log n)$. The runtime was less than 11 seconds, even when datasets formed by about 10 million instances and with more than 10 dimensions are used.

Keywords: transfer function. transfer function specification. clustering. dimensional reduction.

Lista de ilustrações

Figura 1 – Configurações de <i>voxel</i>	20
Figura 2 – <i>Pipeline</i> de visualização volumétrica.	21
Figura 3 – Volume de um pé humano e superfície extraída a partir do volume. . .	23
Figura 4 – Configurações geométricas básicas de ajuste de polígonos em células vo- lumétricas no algoritmo <i>Marching Cubes</i> quando apenas transformações de rotação devem ser realizadas.	24
Figura 5 – <i>Ray Casting</i>	24
Figura 6 – Tecidos biológicos de um pé humano. De cima para baixo: funções de transferência especificadas (cor e opacidade) e estruturas reveladas. . .	26
Figura 7 – Comparação entre as imagens geradas por uma função de transferência 1D baseada em intensidade e uma função de transferência 2D, cujo domínio é composto por intensidade e magnitude de gradiente.	29
Figura 8 – Estruturas de uma cabeça humana reveladas por uma função de trans- ferência baseada em curvatura.	30
Figura 9 – Função de transferência baseada em distância aplicada em um <i>dataset</i> vascular. Regiões enfatizadas: tumor vermelho [0mm, 3mm], vasos san- guíneos laranjas [-15mm, 0mm], vasos sanguíneos amarelos [-30mm, -15mm] e vasos sanguíneos verdes [-45mm, -30mm].	30
Figura 10 – Imagens de um tecido cerebral. (a) Função de transferência 1D baseada em intensidade. (b) Funções de transferência 1D, na qual a cor é mapeada pelo tamanho e a opacidade é mapeada pela intensidade. (c) Função de transferência baseada em tamanho, na qual a opacidade é mapeada pela combinação entre a intensidade e o tamanho.	31
Figura 11 – Imagens geradas por funções de transferência baseadas em forma aplica- das no <i>datasets</i> de angiogramas. As miniaturas são geradas por funções de transferência 1D baseadas em intensidade.	31
Figura 12 – Imagens da fatia dorsal de um abdômen humano geradas pelas seguintes funções de transferência: 2D baseada em gradiente (superior esquerda), 1D baseada em intensidade (superior direita) e 2D baseada em textura (inferior).	32
Figura 13 – Interface padrão para a especificação de funções de transferências 1D. .	33
Figura 14 – Interface padrão com histograma unidimensional para especificação de uma função de transferência de opacidade ¹	34
Figura 15 – Representação de função de transferência unidimensional de opacidade em gráficos de linhas em escala linear e logarítmica.	35

Figura 16 – Interface de histograma multidimensional e <i>widgets</i> em uma função de transferência 2D baseada em intensidade e gradiente.	35
Figura 17 – <i>Design Galleries</i>	36
Figura 18 – Especificação de funções de transferência baseada em desenho. Da esquerda para a direita: fatia do volume com desenhos e imagem resultante da função de transferência especificada.	37
Figura 19 – Técnica de especificação de funções de transferência baseada em contorno aplicado no <i>dataset</i> de uma mão humana.	37
Figura 20 – Exploração volumétrica de uma cabeça humana em um histograma LH.	38
Figura 21 – Especificação de funções de transferência baseadas em visibilidade.	39
Figura 22 – Interface que combina coordenadas paralelas e gráficos de dispersão para especificação de uma função de transferência multidimensional em um volume multivariado.	40
Figura 23 – Características identificadas em um dente humano. Atributos do domínio: intensidade, magnitude de gradiente, gradiente de segunda ordem, entropia, média, desvio padrão e assimetria.	41
Figura 24 – Dendrograma modificado para a exploração do volume de uma cabeça humana.	41
Figura 25 – Exemplo de interação do usuário para exploração do volume por meio da abordagem proposta.	45
Figura 26 – Visão geral das atividades compreendidas no método de classificação de características volumétricas.	48
Figura 27 – Lei dos Cossenos e Teorema de Pitágoras para projeção de um ponto p no hiperplano ortogonal ao segmento de reta $\overline{O_a O_b}$	50
Figura 28 – Exemplo de execução do DBSCAN.	52
Figura 29 – Vizinhaça de uma célula central.	56
Figura 30 – Esquema de especificação inicial da função de transferência.	61
Figura 31 – <i>Lookup Table</i> para armazenamento da função de transferência especificada.	61
Figura 32 – Gráfico de dispersão de características e exploração do volume de um pé. Linha superior: seleções no gráfico de dispersão de características. Linha inferior: visualização decorrente das seleções.	62
Figura 33 – <i>Dataset</i> da cabeça. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,2$ e $minPts = 4$. Atributos: intensidade, assimetria e magnitude de gradiente.	69
Figura 34 – <i>Dataset</i> da carpa. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $minPts = 5$. Atributos: intensidade e assimetria.	70
Figura 35 – <i>Dataset</i> do dente. Configuração de parâmetros: $\beta = 0,75$, $\varepsilon = 0,175$ e $minPts = 5$. Atributos: intensidade, assimetria, magnitude de gradiente e suavidade relativa.	71

Figura 36 – <i>Dataset</i> do pé. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,15$ e $minPts = 4$. Atributos: intensidade, assimetria e entropia.	72
Figura 37 – <i>Dataset</i> da cabeça e 14 características classificadas a partir da intensidade. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $minPts = 4$. . .	74
Figura 38 – <i>Dataset</i> da carpa e 14 características classificadas a partir da intensidade. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $minPts = 4$	75
Figura 39 – <i>Dataset</i> do dente e 14 características classificadas a partir da intensidade. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $minPts = 4$	76
Figura 40 – <i>Dataset</i> do dente e 14 características classificadas a partir da intensidade. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $minPts = 4$	77

Lista de tabelas

Tabela 1	– Exemplo de <i>Lookup Table</i> para uma função de transferência 1D de opacidade.	32
Tabela 2	– Especificação dos <i>datasets</i>	64
Tabela 3	– Operadores de extração de características.	65
Tabela 4	– <i>Ranking</i> de atributos candidatos à composição da assinatura dos <i>voxels</i> para $\alpha = 0,5$	68
Tabela 5	– Tempo de execução do método de classificação de características para os <i>datasets</i> da cabeça, carpa, dente e pé com os atributos dos experimentos da Subseção 5.2.1.	72
Tabela 6	– Tempo de execução do método de classificação de características para os <i>datasets</i> da cabeça, carpa, dente e pé com 11 atributos volumétricos.	73

Lista de algoritmos

1	Heurística para geração do <i>ranking</i> de atributos candidatos.	47
2	Escolha de pivôs do <i>Fastmap</i>	50
3	Rotina principal do DBSCAN.	53
4	Função EXPANDCLUSTER.	54
5	Construção da grade 2D.	56
6	Determinação de <i>core-points</i>	57
7	Determinação de <i>border-points</i> e <i>noises</i>	58
8	Seleção de representativos.	59
9	Identificação de subcaracterísticas em um agrupamento.	60

Lista de abreviaturas e siglas

1D	Unidimensional
2D	Bidimensional
3D	Tridimensional
20D	Icosadimensional
CPU	<i>Central Processing Unit</i> (Unidade Central de Processamento)
DBSCAN	<i>Density-Based Spatial Clustering of Applications with Noise</i> (Agrupamento Espacial Baseado em Densidade de Aplicações com Ruído)
DVR	<i>Direct Volume Rendering</i> (Visualização Direta de Volumes)
FT	Função de Transferência
FTAD	Função de Transferência de Alta Dimensionalidade
GPU	<i>Graphic Processing Unit</i> (Unidade de Processamento Gráfico)
LH	<i>Low-High</i> (Baixo-Alto)
MD	Multidimensional
MDS	<i>Multidimensional Scaling</i> (Escalonamento Multidimensional)
PC	<i>Parallel Coordinates</i> (Coordenadas Paralelas)
PCA	<i>Principal Component Analysis</i> (Análise de Componentes Principais)
SF	<i>Surface Fitting</i> (Ajuste de Superfícies)
SOM	<i>Self-Organizing Map</i> (Mapa Auto Organizável)
SSS	<i>Sparse Spatial Selection</i> (Seleção Esparsa Espacial)
WYSIWYG	<i>What You See Is What You Get</i> (O Que Se Vê É o Que Se Obtém, tradução nossa)

Sumário

1	INTRODUÇÃO	16
1.1	Considerações Iniciais	16
1.2	Objetivos e Contribuições	16
1.3	Organização do Texto	19
2	VISUALIZAÇÃO VOLUMÉTRICA	20
2.1	<i>Pipeline</i> de visualização volumétrica	21
2.2	Algoritmos de Ajuste de Superfícies versus Algoritmos de Visualização Direta e Volumes	22
2.2.1	<i>Marching Cubes</i>	23
2.2.2	<i>Ray Casting</i>	23
2.3	Considerações Finais	25
3	FUNÇÃO DE TRANSFERÊNCIA	26
3.1	Domínio de uma Função de Transferência	27
3.1.1	Atributos Derivados	28
3.2	Implementação de Funções de Transferência	32
3.3	Especificação de Funções de Transferência	33
3.3.1	Abordagens Não Assistidas	34
3.3.2	Abordagens Dirigidas a Imagens	35
3.3.3	Abordagens Dirigidas a Dados	38
3.4	Considerações Finais	41
4	ABORDAGEM DE EXPLORAÇÃO DE VOLUMES MULTIDIMENSIONAIS	43
4.1	Considerações Iniciais	43
4.2	Seleção de Atributos Volumétricos	45
4.3	Método de Classificação de Características Volumétricas	48
4.3.1	Redução Dimensional	49
4.3.2	Agrupamento	51
4.3.3	DBSCAN 2D	53
4.3.4	Seleção de Representativos	58
4.3.5	Classificação de Subcaracterísticas	59
4.4	Interface de Navegação	60
4.4.1	Especificação Inicial da Função de Transferência e <i>Lookup Table</i>	60
4.4.2	Navegação e Interação com o Volume	61

4.5	Considerações Finais	63
5	RESULTADOS	64
5.1	Metodologia	64
5.1.1	Aspectos de Implementação	66
5.1.2	<i>Design</i> dos Experimentos	66
5.2	Experimentos	67
5.2.1	Espaços Multidimensionais	67
5.2.2	Espaços Unidimensionais	70
5.2.3	Tempo de execução	71
5.3	Discussão	73
5.3.1	Seleção de Atributos	73
5.3.2	Exploração do Volume e Configuração de Parâmetros	78
5.4	Considerações Finais	79
6	CONCLUSÃO	80
6.1	Contribuições e Limitações	80
6.2	Trabalhos Futuros	81
	REFERÊNCIAS	82

1 Introdução

1.1 Considerações Iniciais

Visualização volumétrica consiste em um conjunto de técnicas que disponibiliza recursos para a exploração do interior de objetos tridimensionais. Instrumentos de investigação construídos sobre esses mecanismos têm sido recorrentemente utilizados por diferentes áreas da ciência e da indústria, visto que modelos provenientes tanto de dados amostrados quanto de dados sintéticos podem ser manipulados (KAUFMAN, 1990; ELVINS, 1992; KAUFMAN, 1996; BEYER; HADWIGER; PFISTER, 2015).

Visualização Direta de Volumes (DVR, do inglês *Direct Volume Rendering*) é uma categoria-chave de visualização volumétrica que se destaca por produzir imagens de alta qualidade, propiciando que até mesmo pequenos detalhes sejam revelados (ELVINS, 1992). Volumes são em geral dispostos em grades 3D constituídas por partículas que interagem com a luz (MAX, 1995). De acordo com Kindlmann (2002), funções de transferência (FTs) são fundamentais ao processo de DVR, visto que desempenham dois papéis cruciais para a exploração de um volume: mapear dados volumétricos para propriedades visuais (e.g., cor e opacidade) e classificar objetos descritos em tais dados.

É invariavelmente necessário editar FTs para modificar como cada uma das partículas da grade volumétrica se comporta e, assim, visualizar, a cada ajuste, diferentes características inseridas em um volume (KÖNIG; GRÖLLER, 2001). Contudo, construir e especificar uma FT não é uma tarefa simples (KINDLMANN, 2002; LJUNG et al., 2016). No Simpósio de Visualização Volumétrica de 1992, Pat Hanrahan apontou que os tópicos acerca de FTs são um dos dez maiores desafios de visualização volumétrica (PFISTER et al., 2001). Mais de duas décadas se passaram e, apesar do elevado esforço da comunidade científica, FTs ainda concentram questões em aberto (LJUNG et al., 2016).

1.2 Objetivos e Contribuições

A obtenção de uma boa FT é um dos dilemas centrais de DVR (LJUNG et al., 2016). Por “uma boa função”, entenda-se uma configuração de parâmetros capaz de revelar precisamente detalhes de interesse e ocultar as demais regiões. Ao pensar em uma FT como uma função matemática no sentido estrito, é fácil constatar que domínios de dimensionalidades arbitrárias podem ser manipulados. No que diz respeito a tal aspecto, as funções podem ser divididas em dois grupos principais: unidimensionais e multidimensionais. FTs unidimensionais têm uma limitação bem conhecida: produzem classificações imprecisas

quando há objetos distintos que são envolvidos por um mesmo intervalo de valores escalares (KINDLMANN; DURKIN, 1998). Espaços multidimensionais buscam remediar essa deficiência, mas são mais trabalhosos de se editar (KINDLMANN, 2002). A especificação de FTs é, em si, uma tarefa já não intuitiva em espaços unidimensionais que tende a se tornar mais complexa à medida que atributos são adicionados ao domínio (WANG; ZHAO; KAUFMAN, 2012b). Diante desse cenário, uma grande variedade de técnicas para manipular FTs multidimensionais tem sido proposta (LJUNG et al., 2016).

Um outro desafio é a seleção de atributos volumétricos para construção do domínio da FT. Embora o puro aumento da dimensionalidade seja em princípio favorável à classificação, a natureza de cada atributo volumétrico é na verdade um fator mais importante e que, portanto, deve ser foco de atenção durante o projeto da função (ZHOU; SCHOTT; HANSEN, 2012; CAI et al., 2017). Alguns trabalhos – como é o caso daqueles propostos por Arens e Domik (2010) e Zhou, Schott e Hansen (2012) – já demonstraram que uma configuração de domínio pode isolar mais precisamente detalhes volumétricos do que uma outra configuração com a mesma dimensionalidade, mas com constituição diferente. Embora o problema se estenda a outras áreas da ciência (SAEYS; INZA; LARRAÑAGA, 2007), abordagens para apoiar a definição de domínios ainda são incipientes no contexto de FTs (ZHOU; SCHOTT; HANSEN, 2012). O aumento não assistido da dimensionalidade pode produzir ruídos e ser responsável por um efeito contrário à separabilidade de características identificadas em um *dataset* (INDYK; MOTWANI, 1998; SAEYS; INZA; LARRAÑAGA, 2007; ZHOU; SCHOTT; HANSEN, 2012). O referido problema é comum a diferentes áreas de aplicação e popularmente conhecido como Maldição da Dimensionalidade (do inglês *Curse of Dimensionality*) (INDYK; MOTWANI, 1998).

Técnicas de DVR são naturalmente custosas (KAUFMAN, 1996). Apesar dos recentes avanços em tecnologias de hardware – principalmente processamento paralelo em CPUs e GPUs – a crescente utilização de *datasets* maiores e complexos mantém esse problema em discussão. Ljung et al. (2016) indicam que a escalabilidade das técnicas considerando os aspectos de tamanho e dimensionalidade dos dados é uma questão atual e futura.

A ausência de um método capaz de lidar de forma efetiva com todas as questões enunciadas motivou o desenvolvimento deste trabalho. A abordagem proposta abrange todo escopo discutido, focando na edição de FTs e na exploração de volumes. Nossa abordagem é organizada nos três seguintes componentes:

- uma heurística que assiste a seleção de atributos para composição do domínio;
- um método semiautomático que classifica características volumétricas e produz uma especificação inicial da função de transferência;

- uma interface de usuário baseada em um gráfico de dispersão (do inglês *scatter plot*) de características que disponibiliza um espaço simplificado para exploração do volume.

Zhou, Schott e Hansen (2012) introduziram uma técnica que analisa entropia e correlação de um conjunto de atributos volumétricos para indicar o par de atributos que em tese fornece uma classificação mais bem-sucedida de características do *dataset* explorado. Neste trabalho, adotamos correlação múltipla para estender a ideia de Zhou, Schott e Hansen (2012) de modo a gerar um *ranking* de atributos que são os candidatos mais cotados para aumentar a efetividade da FT. A premissa não é trazer uma solução direta para o problema, mas tornar essa etapa do processo de exploração volumétrica menos trabalhosa ao usuário, principalmente quando não há nenhum conhecimento prévio sobre os dados.

O método de classificação de características e a interface de usuário são intimamente ligados a toda a abordagem de exploração. O método combina redução dimensional, agrupamento e seleção de representativos para construção de um gráfico de dispersão de características. O gráfico é formado por agrupamentos de pontos que representam partes (conjunto de *voxels*) do volume a ser explorado. Cada agrupamento é a princípio um material, estrutura ou qualquer outro artefato que foi descoberto pelo método. Para navegar pela interface, o usuário deve selecionar os agrupamentos desejados. Quando isso acontece, a região correspondente no volume é simultaneamente revelada em uma janela de renderização. A interface também permite que diferentes ajustes de cor, opacidade e combinações manuais de agrupamentos sejam realizados.

A contribuição mais evidente deste trabalho é a proposta de uma abordagem de exploração que pode ser, de fato, adotada para tornar a interação menos trabalhosa. O grande número de graus de liberdade típico ao processo de especificação de FTs multidimensionais (CAI et al., 2017) é substituído pela exploração de um gráfico de dispersão de características. A despeito da definição de parâmetros no método de classificação e da técnica de seleção de atributos volumétricos, há uma vigorosa redução do trabalho necessário. Em poucas palavras, a abordagem tem por objetivo abstrair o processo de exploração de dados volumétricos complexos em relação tanto à especificação de FTs quanto à investigação de características. Adicionalmente, manipular de modo efetivo as regiões identificadas.

Ainda é importante destacar que toda a abordagem é escalável. Aqui, esse conceito se alinha ao definido por Ljung et al. (2016) que descreve escalabilidade como a capacidade de manipular de forma eficiente *datasets* cada vez maiores (número de *voxels*) e complexos (dimensionalidade). Independentemente dos parâmetros informados, os algoritmos implementados têm uma complexidade de tempo de $O(n \log n)$. Em todos os experimentos, o

método de classificação combinado à construção da interface consumiram sempre menos do que 11 segundos em execução sequencial para *datasets* com até 11 milhões de *voxels*. Quando um *dataset* formado por mais de 33 milhões de *voxels* e com mais de 10 atributos volumétricos foi analisado, o tempo total de execução foi de aproximadamente 1 minuto e 40 segundos. Em todos os casos, nenhum pré-processamento para preparação da grade volumétrica por meio da remoção de *voxels* é realizado.

1.3 Organização do Texto

Além desta introdução, este documento está organizado em outros cinco capítulos. Conceitos-base de visualização volumétrica são sumariamente descritos no Capítulo 2. A revisão da literatura continua no Capítulo 3, onde são apresentados aspectos relacionados ao projeto e à especificação de FTs. O Capítulo 4 descreve a abordagem de exploração volumétrica proposta. Os resultados são discutidos no Capítulo 5. Finalmente, o Capítulo 6 conclui este trabalho.

A fim de evitar qualquer dúvida durante a leitura deste documento, é importante destacar que o termo “característica” é empregado em dois contextos distintos. Quando nos referimos à extração, a expressão “característica” tem significado de atributo. Como a definição é amplamente difundida nas áreas de processamento de imagens e reconhecimento de padrões, optamos por mantê-la. Entretanto, a expressão “característica” também é utilizada para se referir a um aspecto (e.g., material, estrutura ou objeto) inserido em alguma porção do volume.

2 Visualização Volumétrica

Visualização é um termo condicionado à percepção humana e ao reconhecimento de padrões (MCCORMICK, 1987). Visualização volumétrica é uma subárea da visualização científica que emprega recursos de processamento gráfico para permitir a exploração do interior de volumes (KAUFMAN, 1990; BRODLIE et al., 2012). Técnicas de visualização volumétrica fornecem ferramentas de trabalho para diferentes áreas da ciência, tais como: Biologia, Engenharia Mecânica, Geologia, Química e particularmente Medicina (KAUFMAN, 1990; ELVINS, 1992).

Dados volumétricos armazenam informações que descrevem materiais e regiões 3D. Volume é uma entidade lógica constituída por um conjunto de propriedades organizadas em uma grade de células volumétricas (KAUFMAN, 1996). Um valor escalar ou um vetor de valores escalares é associado a cada posição espacial. Quando todas as células possuem o mesmo tamanho, a mesma forma geométrica e são uniformemente dispostas em relação aos eixos do espaço 3D, a grade volumétrica é classificada como regular. Células de uma grade regular são denominadas *voxels*. Tais estruturas assumem dois tipos de configuração: hexaedros em torno de um valor amostrado ou hexaedros cujos vértices são valores amostrados (ELVINS, 1992). Ambas as configurações são apresentadas na Figura 1.

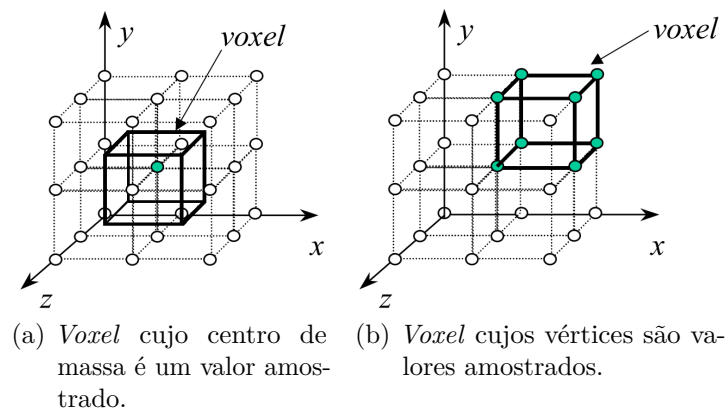


Figura 1 – Configurações de *voxel*.

Fonte: adaptado de Paiva, Seixas e Gattass (1999, p. 2).

Elvins (1992) classifica os algoritmos de visualização volumétrica em duas categorias principais: algoritmos de Ajuste de Superfícies (SF, do inglês *Surface Fitting*) e algoritmos de Visualização Direta de Volumes (DVR, do inglês *Direct Volume Rendering*). Em síntese, algoritmos da primeira categoria utilizam uma representação geométrica intermediária, enquanto que algoritmos da segunda categoria atuam diretamente sobre a grade volumétrica. Uma comparação mais detalhada entre ambas as categoriais é apresentada na Seção 2.2.

2.1 Pipeline de visualização volumétrica

Algoritmos de visualização volumétrica podem ser sistematizados em uma sequência de quatro etapas básicas chamada *pipeline* de visualização volumétrica. A Figura 2 apresenta um esquema que ilustra o *pipeline*.

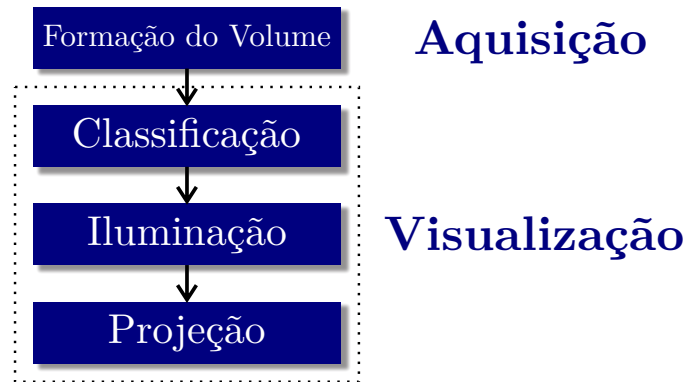


Figura 2 – Pipeline de visualização volumétrica.

Fonte: adaptado de Paiva, Seixas e Gattass (1999, p. 4).

A etapa de formação do volume não é, de fato, parte do processo de visualização, apesar de ser incumbida da aquisição de dados (ELVINS, 1992). Dados volumétricos podem ser sintetizados por simulações computacionais e modelos matemáticos, ou amostrados a partir de objetos do mundo real (KAUFMAN, 1996). Procedimentos de amostragem tipicamente utilizam dispositivos de varredura, como: *scanners*, sismógrafos e equipamentos de Imagem por Ressonância Magnética ou de Tomografia Computadorizada (ELVINS, 1992). O processo de aquisição usualmente fornece apenas informações da densidade dos materiais, mas outras informações (e.g., magnitude de fluxo e energia) também podem ser fornecidas. Dados volumétricos são chamados de “multivariados” quando mais do que um atributo é obtido (STOMPEL; LUM; MA, 2002). Uma vez adquiridos, os dados são interpolados para formação do volume.

O processo de visualização se inicia na etapa de classificação. A etapa patrocina diretamente a exploração do volume, disponibilizando mecanismos para revelar diferentes materiais e artefatos durante a interação do usuário. Os recursos disponibilizados compreendem sobretudo a extração de superfícies de regiões em SF e a especificação de funções de transferência em DVR.

Na etapa de iluminação, modelos tradicionais de sombreado (e.g., Gouraud e Phong) são aplicados no volume com intuito de conferir realismo às estruturas e regiões visíveis (LEVOY, 1988; ELVINS, 1992).

Por fim, a etapa de projeção é responsável por gerar a imagem que será visualizada. Algoritmos de SF delegam essa tarefa a técnicas tradicionais de renderização de polígonos (e.g. *Back-Face Culling* e *Z-buffer*). Distintivamente, algoritmos de DVR se baseiam no

processo físico de emissão-absorção de luz para mapear as informações volumétricas no plano de imagem por meio de um processo de composição.

2.2 Algoritmos de Ajuste de Superfícies versus Algoritmos de Visualização Direta e Volumes

Algoritmos de Ajustes e Superfícies atuam sobre representações geométricas computadas a partir do modelo original. Inicialmente, o usuário estabelece um valor de limiarização (do inglês, *threshold*) e, conseqüentemente, um contorno sobre a grade volumétrica. Primitivas geométricas são então ajustadas dentro das regiões delimitadas para extração de uma malha (ELVINS, 1992). Em um último procedimento, técnicas de renderização de polígonos são executadas para visualização das estruturas (ELER, 2006). O baixo custo computacional exigido por tais técnicas permite que, por via de regra, seja mais rápido interagir com ferramentas de SF do que de DVR. Entretanto, a utilização de algoritmos de SF também traz desvantagens. O processo de extração pode produzir artefatos e ruídos não presentes nos dados originais. Há ainda uma evidente perda de informações que é própria a esse tipo de operação. Elvins (1992) afirma que algoritmos de SF não são adequados a todas as situações de exploração volumétrica.

Algoritmos de Visualização Direta de Volumes dispensam a utilização de uma representação intermediária, projetando diretamente as informações armazenadas no plano de imagem. Apesar do maior custo computacional, o processo preserva todas as características e é, por isso, capaz de gerar imagens de maior qualidade (KAUFMAN, 1996). O volume é tratado como um grupo organizado de partículas que simultaneamente absorvem e refletem parte da luz que as atravessa. Algoritmos de DVR são frequentemente utilizados em situações que exigem visualizações ricas em detalhes (ELVINS, 1992). Um exemplo típico ocorre quando é necessário visualizar objetos amorfos, como é o caso de nuvens e gases.

Independentemente do tipo de renderização volumétrica, a interação com um volume envolve a modificação de parâmetros de cena ou a manutenção da visibilidade de materiais. Algoritmos de SF lidam de modo distinto com cada uma das situações. Quando a cena é modificada, apenas a etapa de renderização de polígonos é executada. Por outro lado, é previamente necessário extrair superfícies para descobrir estruturas que não são visíveis em um primeiro contato. Algoritmos de DVR procedem do mesmo modo para qualquer que seja a forma de interação. Nessa última categoria, a visibilidade de uma região ou material depende da especificação de funções de transferência.

Uma comparação entre o volume de um pé humano e a superfície extraída a partir desse volume é apresentada na Figura 3. Ambas as imagens foram geradas com o software

Kitware ParaView¹.

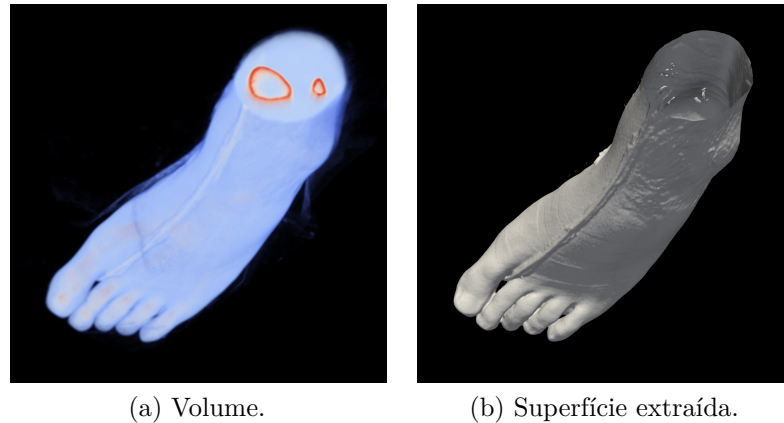


Figura 3 – Volume de um pé humano e superfície extraída a partir do volume.

2.2.1 *Marching Cubes*

Marching Cubes é uma técnica de SF, originalmente proposta por Lorensen e Cline (1987), que tem por ideia central extrair isosuperfícies – superfícies associadas a um mesmo intervalo de valores amostrados.

O usuário inicialmente informa isovalores para delimitar regiões do volume. Dados amostrados armazenados nos vértices de uma célula determinam se o respectivo vértice pertence ou não à isosuperfície que será computada. Após encontrar todos os vértices que integram a estrutura, polígonos são ajustados dentro de cada célula interceptada. Como células volumétricas são comumente hexaedros, há ao todo 256 (8^3) configurações geométricas possíveis. Na prática, a maioria das implementações reduz o número de configurações a um conjunto menor de possibilidades (NEWMAN; YI, 2006). Uma abordagem frequentemente empregada considera apenas as 23 configurações que são apresentadas na Figura 4. Na abordagem, as demais geometrias são obtidas por meio de transformações de rotação. Já com os polígonos ajustados, o *Marching Cubes* examina as células interceptadas, combinando-as para a extração da isosuperfície resultante.

2.2.2 *Ray Casting*

Ray Casting é uma das técnicas de DVR mais difundidas na literatura. As primeiras versões do algoritmo – os trabalhos de Tuy e Tuy (1984), e Levoy (1988) – datam da década de 1980. Uma variedade de otimizações foi proposta desde então.

A Figura 5 apresenta o funcionamento básico do algoritmo. Antes de tudo, um plano de imagem é posicionado entre o observador e a grade volumétrica. Um raio é então

¹ Kitware ParaView. Disponível em: <<https://www.paraview.org/>>.

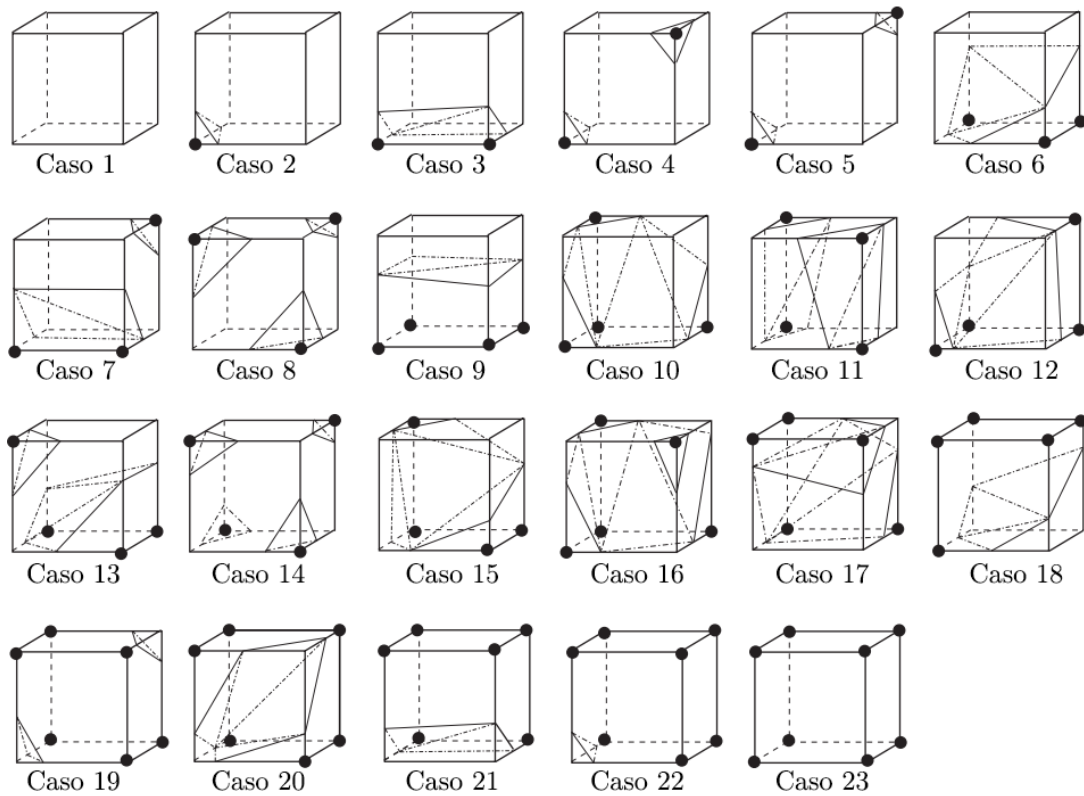


Figura 4 – Configurações geométricas básicas de ajuste de polígonos em células volumétricas no algoritmo *Marching Cubes* quando apenas transformações de rotação devem ser realizadas.

Fonte: Newman e Yi (2006, p. 857, tradução nossa).

disparado a partir de cada pixel do plano em direção ao volume. Os raios atravessam a grade volumétrica e, de modo iterativo, atingem diferentes amostras ao longo do percurso. Cada amostra atingida contribui com cor e opacidade. As contribuições são acumuladas até que o caminhar do raio cesse e a cor computada seja enfim atribuída ao pixel de origem.

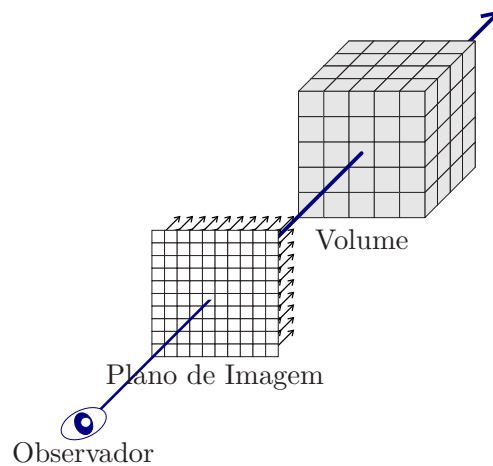


Figura 5 – *Ray Casting*.

A equação que descreve a acumulação de cor e opacidade ao longo do caminharmento de um raio é denominada equação de renderização volumétrica (SABELLA, 1988). Baseada no modelo físico de emissão-absorção de luz em materiais, a equação é definida por Max (1995) como:

$$I = \int_a^b \mathbf{q}(s) e^{\int_a^s K(\mu) d\mu} ds, \quad (2.1)$$

onde I é a luz resultante após o caminharmento de um ponto a até um ponto b , $\mathbf{q}(s)$ é a contribuição de luz incidente na amostra s e $K(\mu)$ é a função que descreve o transporte de luz através de um material.

Em versão discreta, a Equação 2.1 pode ser reescrita por meio de operações de composição de cor e opacidade. Para cada pixel, a equação resultante assume a seguinte forma:

$$I = \sum_{i=1}^n C_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \quad (2.2)$$

onde C_i é a cor e α_i é a opacidade que são absorvidas por um material volumétrico na posição atingida pelo raio (LJUNG et al., 2016). Os valores de C_i e α_i são obtidos a partir de funções de transferência, sendo assim suscetíveis a modificações.

Durante o caminharmento, um ponto atingido pode não coincidir perfeitamente com um ponto de amostra. Quando isso acontece, uma contribuição aproximada deve ser encontrada. Entre as técnicas utilizadas para esse propósito, destacam-se: vizinho mais próximo e interpolação trilinear. A primeira escolhe o ponto mais próximo da posição atingida. Já a segunda, analisa a proximidade com as amostras da vizinhança para combinar linearmente a contribuição resultante. Em comparação à escolha do vizinho mais próximo, a interpolação produz imagens de maior qualidade, mas demanda um maior tempo de execução.

2.3 Considerações Finais

Embora não seja um tópico de estudo surgido recentemente, visualização volumétrica sempre teve sua popularidade comprometida pelo elevado custo computacional exigido. Nas últimas duas décadas, o crescente desenvolvimento de tecnologias de hardware – e.g., computação distribuída, CPUs multinúcleo e principalmente GPUs – e a introdução de novas técnicas têm propiciado a transposição dessa barreira (BEYER; HADWIGER; PFISTER, 2014). Todavia, em face à consolidação desse cenário positivo, o aumento do tamanho e da complexidade dos *datasets* volumétricos têm continuamente impulsionado esse campo de pesquisa (ENCARNACAO et al., 2015).

3 Função de Transferência

Funções de transferência desempenham dois papéis fundamentais durante o processo de exploração em Visualização Direta de Volumes:

- mapear informações armazenadas nos dados para propriedades visuais, principalmente cor e opacidade;
- classificar materiais volumétricos, mediante a definição de parâmetros.

FTs podem ser especificadas para revelar diferentes estruturas, materiais ou demais artefatos durante o processo de exploração volumétrica (KINDLMANN; DURKIN, 1998; KNISS; KINDLMANN; HANSEN, 2002). A título de exemplo, a Figura 6 apresenta imagens de um pé humano geradas por duas especificações distintas da mesma função. Uma primeira configuração propicia que a cútis (veja a Subfigura 6a) seja visualizada, ao passo que uma variação dos parâmetros revela o tecido ósseo (veja a Subfigura 6b).

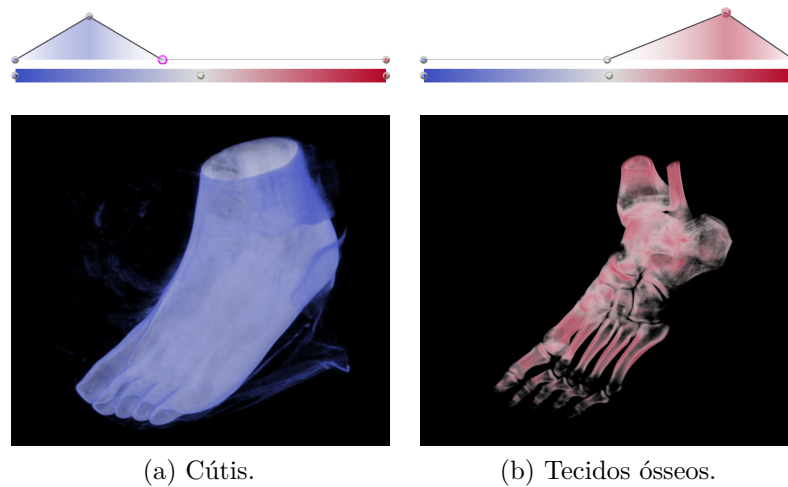


Figura 6 – Tecidos biológicos de um pé humano. De cima para baixo: funções de transferência especificadas (cor e opacidade) e estruturas reveladas.

De acordo com König e Gröller (2001), uma FT τ é formalmente descrita em função de um domínio de atributos volumétricos X e um contradomínio de propriedades visuais V , como estabelecido a seguir:

$$\tau : X_1 \times X_2 \times \dots \times X_n \rightarrow V_1 \times V_2 \times \dots \times V_m, \quad (3.1)$$

onde n é a dimensão do domínio e m é a dimensão do contradomínio.

O modelo apresentado na Equação 3.1 permite formular um exemplo algébrico de especificação de uma FT de opacidade como:

$$\alpha(x) = \begin{cases} 0 & \text{se } x \leq Z_0 \\ \frac{x}{I} & \text{se } Z_0 < x \leq Z_1, \\ 0,5 & \text{se } x > Z_1 \end{cases} \quad (3.2)$$

onde $\alpha(x)$ é opacidade, Z_0 e Z_1 são constantes, x é o valor de intensidade armazenado em uma amostra qualquer e I , a intensidade máxima encontrada no domínio.

A utilização de FTs remonta aos primeiros algoritmos de Visualização Direta de Volumes – como é o caso dos trabalhos de [Levoy \(1988\)](#) e [Drebin, Carpenter e Hanrahan \(1988\)](#) – ainda que o termo “função de transferência” não tenha sido empregado naquele contexto. Desde então, o campo de pesquisa continua ativo como um dos principais tópicos de visualização volumétrica ([LJUNG et al., 2016](#)). Abordagens a respeito do tema se concentram em dois pontos-chave: natureza físico-matemática dos atributos volumétricos e métodos de especificação. Neste capítulo, direcionamos nossa discussão a esses dois pontos e também a alguns tópicos relacionados.

3.1 Domínio de uma Função de Transferência

FTs são, em suas formas mais simples, unidimensionais, i.e., definidas em função de uma única variável de entrada ([KINDLMANN; DURKIN, 1998](#)). Na maioria dos *datasets*, essa variável assume valores de intensidade. Apesar da ampla adoção de domínios unidimensionais, utilizar um único atributo pode ser prejudicial ao processo de classificação ([KINDLMANN, 2002; LJUNG et al., 2016](#)). FTs unidimensionais não identificam com precisão objetos que, embora espacialmente distintos, são “ativados” por um mesmo intervalo de valores ([PFISTER et al., 2001; ZHOU; SCHOTT; HANSEN, 2012](#)). FTs multidimensionais se servem de atributos volumétricos adicionais para combater essa deficiência ([KNISS; KINDLMANN; HANSEN, 2001; ZUDILOVA-SEINSTRA; ADRIAANSEN; LIERE, 2008](#)). Domínios multidimensionais são comumente bidimensionais ou tridimensionais, mas uma maior quantidade de atributos também pode ser adicionada. Quando isso acontece, adota-se a designação funções de transferência de alta Dimensionalidade (FTAD).

Na prática, a dimensionalidade do domínio não é, em si, responsável pela capacidade de classificação, mas sim a natureza dos atributos volumétricos combinados no espaço da função ([ARENS; DOMIK, 2010; ZHOU; SCHOTT; HANSEN, 2012](#)). Alguns autores denominam essa combinação como “assinatura dos *voxels*” ([TENGINAKAI; LEE; MACHIRAJU, 2001; NAM; MAURER; MUELLER, 2009](#)). Construir assinaturas que aumentem verdadeiramente o poder de classificação não é uma tarefa simples, já que a

escolha de atributos depende do *dataset* envolvido e das características que se tem interesse identificar (ZHOU; SCHOTT; HANSEN, 2012). O próprio processo de aquisição fornece ocasionalmente mais de um atributo e FTs podem pontualmente atuar sobre esse tipo do domínio. Entretanto, dados volumétricos são, na maior parte das vezes, constituídos apenas por valores escalares de intensidade (LJUNG et al., 2016). Nesses casos, só é possível operar sobre espaços multidimensionais por meio da extração de atributos derivados.

3.1.1 Atributos Derivados

Diferentes atributos volumétricos derivados foram propostos ao longo dos anos. Nesta subseção, apresentamos questões pertinentes à intensidade e a alguns atributos derivados de tal propriedade: gradiente, curvatura, distância, tamanho, forma e textura.

Intensidade é um aspecto vinculado à densidade física de materiais, que serve de dado bruto para a maior parte dos atributos derivados. FTs baseadas em intensidade são sensíveis a discrepâncias de tonicidade e são, melhor dizendo, capazes de delimitar regiões monotônicas (ARENS; DOMIK, 2010).

Magnitude de gradiente é um atributo que, associado à vizinhança de células volumétricas, mede a taxa de variação local de intensidade. O gradiente pode ser interpretado como um vetor $\vec{v}$, cuja direção indica a região de maior variação. O cálculo do atributo é formalmente expresso pela primeira derivada direcional da função de intensidade. Considerando f como a distribuição de intensidade, a função gradiente f' é escrita como:

$$f' = \|\nabla f\| \quad (3.3)$$

Não é sempre possível obter o gradiente de forma analítica, já que volumes são formados por elementos discretos. Valores aproximados são frequentemente calculados por operadores de diferenças finitas.

Ao incorporar o gradiente, uma FT se torna capaz de identificar bordas e superfícies, além de regiões monotônicas. A Figura 7 apresenta uma comparação entre as características enfatizadas pela função antes e depois da adição do atributo. A FT 2D revela estruturas (veja a Subfigura 7b) que não são nitidamente visíveis (veja a Subfigura 7a) na imagem produzida pela FT 1D.

A magnitude de gradiente não é, em alguns casos, suficiente para distinguir bordas de regiões volumétricas sobrepostas (KNISS; KINDLMANN; HANSEN, 2001). Diante desse problema, um outro atributo associado ao gradiente pode ser introduzido: a segunda derivada direcional da função de intensidade, também chamada de primeira derivada do gradiente. Tal propriedade é obtida por uma combinação entre a magnitude de gradiente e uma matriz de derivadas parciais, a matriz de Hessian. De acordo com Kniss, Kindlmann

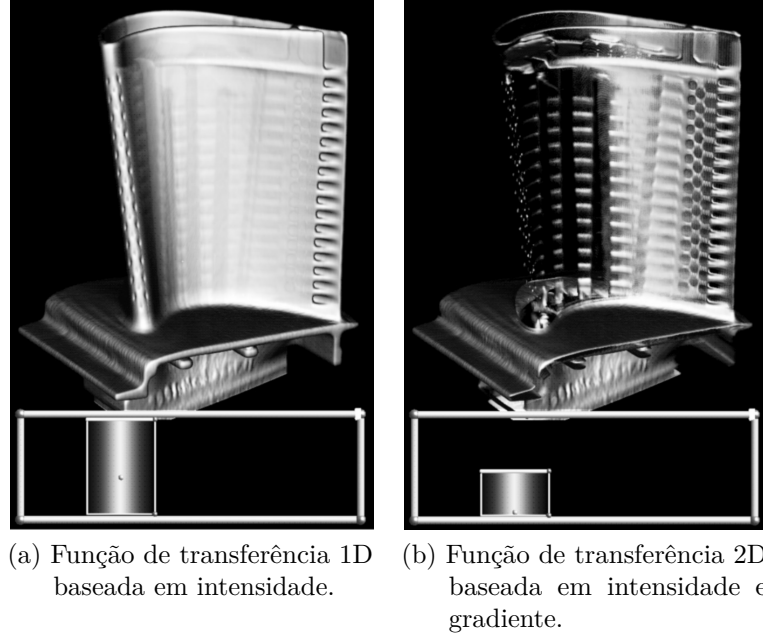


Figura 7 – Comparação entre as imagens geradas por uma função de transferência 1D baseada em intensidade e uma função de transferência 2D, cujo domínio é composto por intensidade e magnitude de gradiente.

Fonte: Kniss, Kindlmann e Hansen (2001, p. 257).

e Hansen (2002), o cálculo dessa combinação é definido como:

$$f'' = \frac{1}{\|\nabla f\|^2} (\nabla f)^T \mathbf{H} f \nabla f, \quad (3.4)$$

onde f'' é a segunda derivada direcional da função de intensidade e $\mathbf{H}$ é a matriz Hessian. Alternativamente, a função f'' pode ser aproximada pelo laplaciano. O operador exige um menor custo computacional, mas leva a resultados menos precisos do que a primeira abordagem.

Curvatura é uma métrica computada por convolução de sinais que descreve contornos de formas locais (ARENS; DOMIK, 2010; LJUNG et al., 2016) e mensura o quanto uma superfície se desvia do aspecto retilíneo. O atributo é comumente utilizado para a geração de visualizações não foto-realísticas (ZHOU; SCHOTT; HANSEN, 2012). Uma curvatura é geralmente derivada de duas magnitudes principais: uma máxima k_1 e uma mínima k_2 (LJUNG et al., 2016). A partir de tais componentes, é possível obter diferentes tipos de curvatura, entre os quais destacam-se: a curvatura da média e a gaussiana (PREIM; BOTH, 2013). A Figura 8 apresenta superfícies detectadas por uma FT baseada em curvatura aplicada no *dataset* de uma cabeça humana.

Distância é um atributo volumétrico que não é diretamente derivado da intensidade, mas calculado sobre partes pré-segmentadas de um volume (TAPPENBECK; PREIM; DICKEN, 2006). FTs baseadas em distância identificam regiões próximas a um local de

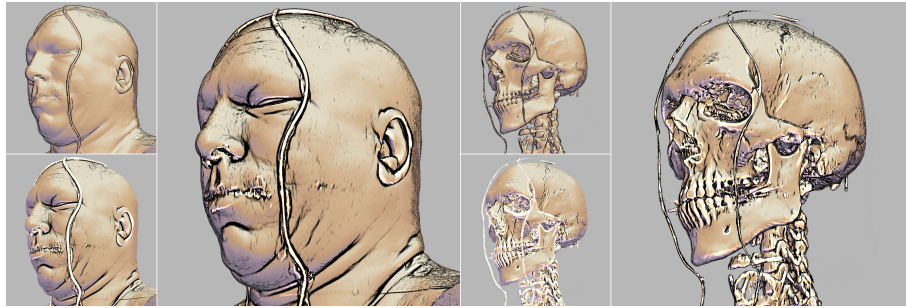


Figura 8 – Estruturas de uma cabeça humana reveladas por uma função de transferência baseada em curvatura.

Fonte: Kindlmann et al. (2003, p. 519).

interesse, como pode ser observado no exemplo da Figura 9, onde camadas adjacentes delimitam vizinhanças esféricas de um tumor em um *dataset* vascular.

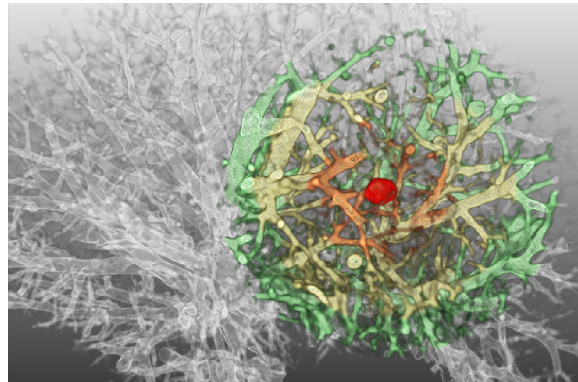


Figura 9 – Função de transferência baseada em distância aplicada em um *dataset* vascular. Regiões enfatizadas: tumor vermelho [0mm, 3mm], vasos sanguíneos laranjas [-15mm, 0mm], vasos sanguíneos amarelos [-30mm, -15mm] e vasos sanguíneos verdes [-45mm, -30mm].

Fonte: adaptado de Tappenbeck, Preim e Dicken (2006, p. 271).

Correa e Ma (2008) adaptaram a ideia de tamanho ao contexto de DVR. Assim como acontece em Visão Computacional, é difícil definir descritores matemáticos que reconheçam variações de tamanho, embora seja visualmente intuitivo interpretar esse conceito (PREIM; BOTH, 2013). Os autores desenvolveram uma técnica que analisa campos multiescalares e esquemas piramidais para categorizar objetos inseridos em um volume. A Figura 10 apresenta imagens de um cérebro humano com aneurisma geradas por FTs baseadas em tamanho e em intensidade.

Forma é um operador que, tal como a distância, é preparado sobre volumes segmentados. Praßni et al. (2010) propuseram uma FT baseada em forma que examina três aspectos para discriminar um objeto: *tubiness* (aspecto cilíndrico), *surfacedness* (aspecto planar) e *blobbiness* (aspecto de bolha). Imagens resultantes de uma FT baseada em forma

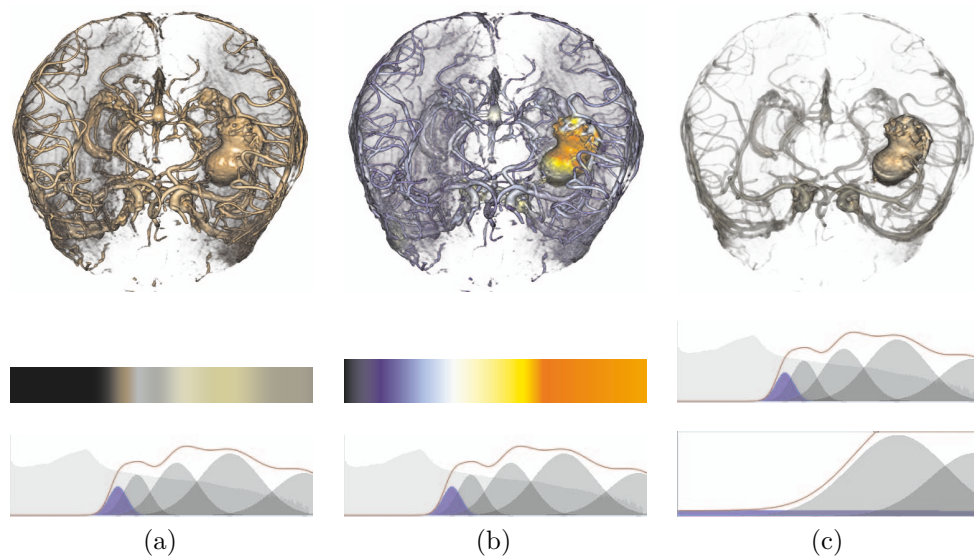


Figura 10 – Imagens de um tecido cerebral. (a) Função de transferência 1D baseada em intensidade. (b) Funções de transferência 1D, na qual a cor é mapeada pelo tamanho e a opacidade é mapeada pela intensidade. (c) Função de transferência baseada em tamanho, na qual a opacidade é mapeada pela combinação entre a intensidade e o tamanho.

Fonte: adaptado de [Correa e Ma \(2008, p. 1381\)](#).

aplicada em *datasets* de angiogramas são apresentadas na Figura 11. Em ambos os volumes visualizados, um nódulo é destacado em relação às demais estruturas.

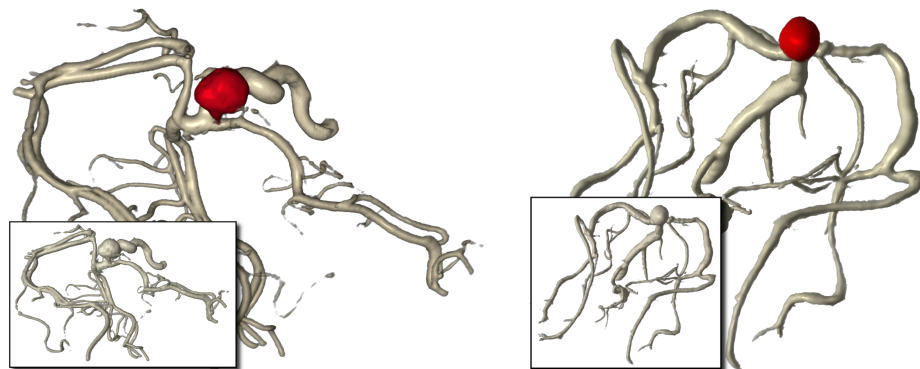


Figura 11 – Imagens geradas por funções de transferência baseadas em forma aplicadas no *datasets* de angiogramas. As miniaturas são geradas por funções de transferência 1D baseadas em intensidade.

Fonte: [Praßni et al. \(2010, p. 16\)](#).

Textura é um conceito oriundo do reconhecimento de padrões que tem sido gradativamente empregado em visualização volumétrica. Computada por descritores estatísticos locais, textura é uma propriedade composta que não é formada por um número fixo de atributos. A Figura 12 apresenta imagens do volume da fatia dorsal de um abdômen

humano geradas pela especificação de três funções distintas: uma FT 2D baseada em gradiente e intensidade, uma FT 1D baseada em intensidade (superior à direita) e uma FT 20D baseada em textura (inferior).

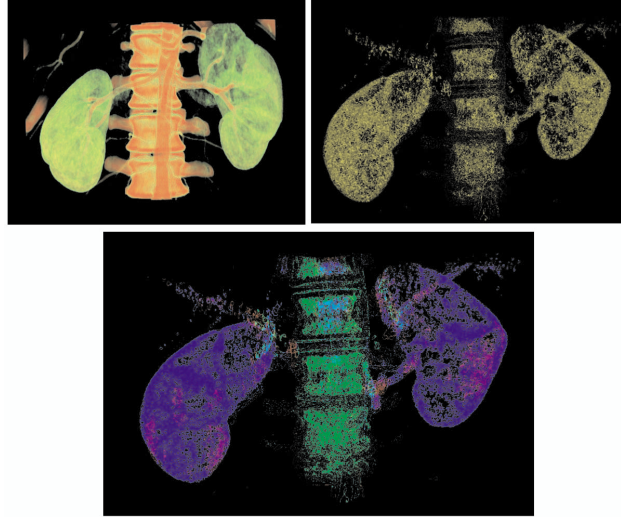


Figura 12 – Imagens da fatia dorsal de um abdômen humano geradas pelas seguintes funções de transferência: 2D baseada em gradiente (superior esquerda), 1D baseada em intensidade (superior direita) e 20D baseada em textura (inferior).

Fonte: Caban e Rheingans (2008, p. 1369).

3.2 Implementação de Funções de Transferência

Lookup Table (LUT) é a abordagem mais disseminada para implementação de FTs especificadas. Do ponto de vista matemático, LUT é uma tabela indexada de discretizações de propriedades visuais e valores escalares de dados volumétricos (PREIM; BOTH, 2013). Usualmente LUTs não são diretamente editadas. A maior parte das ferramentas implementa interfaces gráficas e, somente após a interação do usuário nas interfaces, calcula interpolações dos parâmetros ajustados para então preencher a tabela.

Um exemplo de LUT para uma FT unidimensional de opacidade é apresentado na Tabela 1. Cada linha contém um valor x_i de intensidade e $\alpha(x_i)$ de opacidade.

Tabela 1 – Exemplo de *Lookup Table* para uma função de transferência 1D de opacidade.

x	$\alpha(x)$
0	0,0
1	0,05
2	0,01
$\vdots$	$\vdots$
255	1,0

Em contraste a interoperabilidade entre FTs unidimensionais especificadas e LUTs, é um desafio representar FTs multidimensionais (KNISS; KINDLMANN; HANSEN, 2002). A adição de atributos provoca um grande aumento – em alguns casos, até exponencial – do tamanho da LUT e consequentemente do espaço de armazenamento (KNISS et al., 2003). Ao longo dos anos, esse problema estimulou o desenvolvimento de diferentes estratégias. Algumas implementações fragmentam a estrutura, decompondo a FT especificada em múltiplas funções de menor dimensionalidade. Outras abordagens dispensam a utilização da tabela e armazenam a configuração de forma analítica, calculando a propriedade visual sempre que a informação for requisitada. Ademais, há abordagens que utilizam primitivas gaussianas (KNISS et al., 2003) para compactar o domínio e, assim, o espaço de armazenamento necessário.

3.3 Especificação de Funções de Transferência

Até o momento, salientamos que a habilidade de classificação de uma FT é subordinada à assinatura dos *voxels*. De modo complementar, é adicionalmente necessário especificar a função de forma adequada para visualizar os objetos desejados. Encontrar uma configuração ideal é uma tarefa já não intuitiva e não trivial em espaços unidimensionais, que tende a se tornar mais complexa à medida que a dimensionalidade do domínio aumenta (PINTO; FREITAS, 2007; WANG; ZHAO; KAUFMAN, 2012b).

A Figura 13 apresenta uma interface de usuário típica para a especificação de FTs unidimensionais. Pontos de controle são ajustados para delimitar segmentos de reta (KNISS; KINDLMANN; HANSEN, 2002). A especificação resultante é formada pela organização de tais segmentos em um gráfico de linhas, cujo eixo das abcissas representa a intensidade e o eixo das coordenadas, uma propriedade visual – no caso ilustrado, cor ou opacidade.

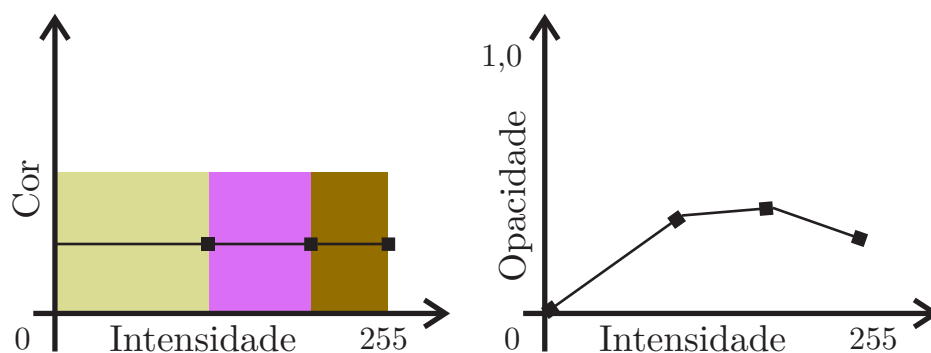


Figura 13 – Interface padrão para a especificação de funções de transferências 1D.

Relativamente ao nível de interação, métodos de especificação são categorizados em: não assistidos, automáticos e semiautomáticos. Métodos da primeira categoria não proveem nenhum auxílio, impelindo ao usuário toda a responsabilidade pelo ajuste de

parâmetros (LJUNG et al., 2016). Em contrapartida, métodos automáticos diminuem o esforço exigido, mas são mais propensos a erros. Não considerar o *feedback* do usuário pode induzir toda exploração do volume a um resultado mal-sucedido (KÖNIG; GRÖLLER, 2001; PFISTER et al., 2001; KINDLMANN, 2002), posto que o reconhecimento de características tem como cerne à percepção humana. A utilização de métodos semiautomáticos desponta como uma importante alternativa em meio a esse cenário (PFISTER et al., 2001). Abordagens semiautomáticas buscam agrupar vantagens de ambos os níveis de automaticidade para diminuir o esforço e, ao mesmo tempo, manter o usuário inserido no processo (PREIM; BOTHA, 2013).

Os tipos de estratégias adotadas para auxiliar o usuário durante a especificação permitem que abordagens não assistidas – tanto automáticas, quanto semiautomáticas – sejam organizadas em: não assistidas, dirigidas a imagens e dirigidas a dados (KINDLMANN, 2002). É importante frisar que não há um limite rígido entre tais grupos, já que diferentes dispositivos são muitas vezes combinados em uma mesma técnica (PRAUCHNER; FREITAS; COMBA, 2005).

3.3.1 Abordagens Não Assistidas

A interface apresentada na Figura 13 é convencionalmente utilizada para a manipulação de FT unidimensional (KNISS; KINDLMANN; HANSEN, 2002). Embora a abordagem propicie uma conversão quase imediata da FT especificada em uma LUT, a forma de interação imposta conduz o usuário a um processo de “tentativa e erro” (PRAUCHNER; FREITAS; COMBA, 2005). Algumas ferramentas adicionam histogramas ao plano de fundo da interface de usuário com o propósito de atenuar esse problema. Um exemplo pode ser observado na Figura 14. Uma outra alternativa envolve a utilização de histogramas editáveis (KNISS; KINDLMANN; HANSEN, 2002).

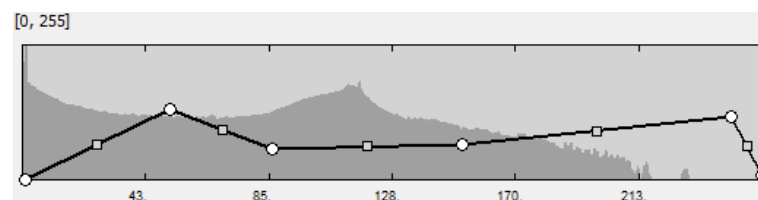


Figura 14 – Interface padrão com histograma unidimensional para especificação de uma função de transferência de opacidade¹.

Potts e Möller (2004) substituíram os gráficos em escala linear comumente utilizados, como os exemplos das Figuras 13 e 14, por gráficos em escala logarítmica. Uma comparação entre ambos os tipos de escala é apresentada na Figura 15.

Uma alternativa não assistida bem difundida para manipular espaços 2D e 3D é apresentada na Figura 16. A interface é formada por um histograma multidimensional, no

¹ Interface de usuário do software Kitware VolView.

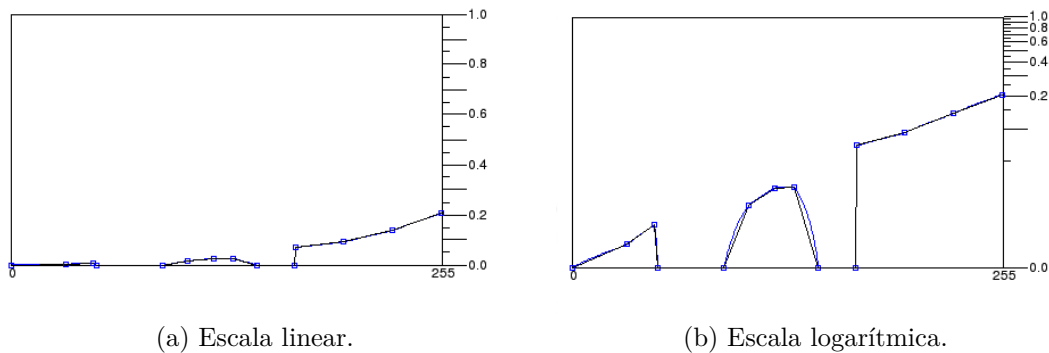


Figura 15 – Representação de função de transferência unidimensional de opacidade em gráficos de linhas em escala linear e logarítmica.

Fonte: Potts e Möller (2004, p. 62-63).

qual *widgets* (primitivas geométricas, tradução nossa) são movidos e redimensionados para exploração do volume.

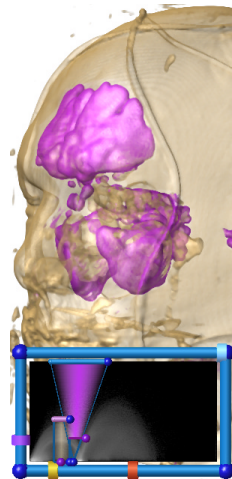


Figura 16 – Interface de histograma multidimensional e *widgets* em uma função de transferência 2D baseada em intensidade e gradiente.

Fonte: adaptado de Kniss, Kindlmann e Hansen (2002, p. 273).

3.3.2 Abordagens Dirigidas a Imagens

Abordagens dirigidas a imagens utilizam fatias do *dataset* ou imagens já renderizadas para guiar o usuário durante a especificação (KINDLMANN, 2002).

He et al. (1996) desenvolveram um dos primeiros trabalhos que se apodera de uma estratégia dirigida a imagens. Inicialmente, um conjunto de FTs especificadas é gerado estocasticamente e aplicado no volume. As imagens resultantes são apresentadas ao usuário, que seleciona as que mais se aproximam do perfil desejado. Quando a seleção termina,

uma nova população estocástica de especificações candidatas é fornecida. O procedimento é repetido até que uma visualização satisfatória seja encontrada.

Design Galleries (MARKS et al., 1997) é uma técnica que generaliza a ideia introduzida por He et al. (1996) a todo o processo de exploração volumétrica, abrangendo inclusive informações de cena (e.g., iluminação e posição do observador). A técnica ainda constrói uma galeria, onde as imagens renderizadas são reduzidas a *thumbnails* (miniaturas de imagem, tradução nossa) e dispostas segundo uma relação de similaridade. Um exemplo do conceito é apresentado na Figura 17. O usuário deve navegar pela galeria e selecionar os *thumbnails* desejados. As especificações cujos *thumbnails* não forem selecionados são substituídas por meio de uma variação da configuração.

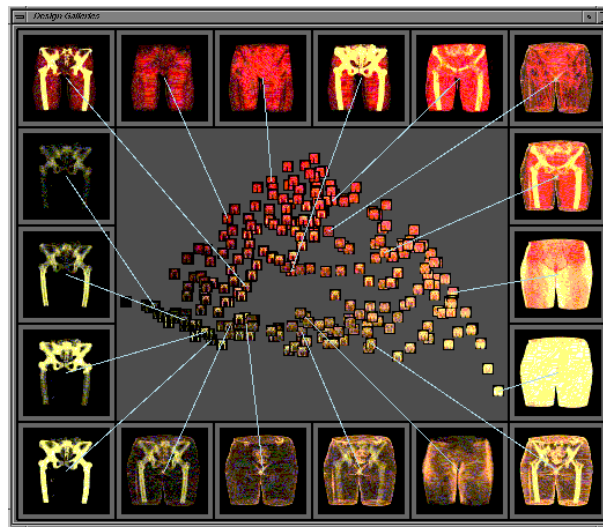


Figura 17 – *Design Galleries*.

Fonte: Marks et al. (1997, p. 399).

Prauchner, Freitas e Comba (2005) propuseram uma abordagem baseada em dois níveis de interação que alia a galeria de *thumbnails* a uma interface não assistida. No primeiro nível de interação, o usuário realiza um ajuste manual, mesmo que grosseiro, da FT. Posteriormente, uma perturbação dos parâmetros produz uma população de especificações. *Thumbnails* decorrentes dessa população são organizadas em galeria. No segundo nível de interação, as especificações produzidas passam a ser manualmente refinadas. Pinto e Freitas (2008) trabalharam na técnica de Prauchner, Freitas e Comba (2005) e adicionaram a proposta original uma árvore de estados que armazena o histórico de edições.

Além das principais estratégias citadas até aqui – concepção estocástica de FTs especificadas e galeria de *thumbnails* – abordagens dirigidas a imagens costumam se apropriar de recursos disponíveis em softwares de edição de imagem, como por exemplo: pincéis, baldes de preenchimento e borrachas (LJUNG et al., 2016).

Tzeng, Lum e Ma (2003) utilizaram uma rede neural para ajustar a FT com base em um esboço desenhado sobre uma visualização do volume. A rede é treinada até que

uma especificação capaz de enfatizar as regiões assinaladas seja encontrada. A Figura 18 apresenta um exemplo de interação, no qual cada esboço desenhado enfatiza uma estrutura diferente após o treinamento.

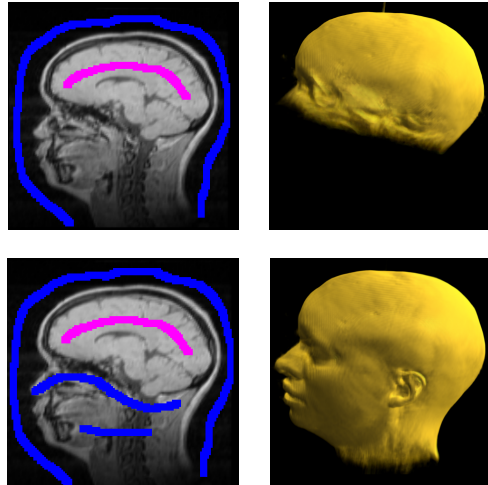


Figura 18 – Especificação de funções de transferência baseada em desenho. Da esquerda para a direita: fatia do volume com desenhos e imagem resultante da função de transferência especificada.

Fonte: adaptado de Tzeng, Lum e Ma (2003, p. 510).

Ropinski et al. (2008) utilizaram contornos para separar porções de um volume antes do processo de especificação. Após isolar a região desejada, o usuário especifica manualmente a FT. Uma sequência de interações que ilustra todo o processo pode ser observada na Figura 19.

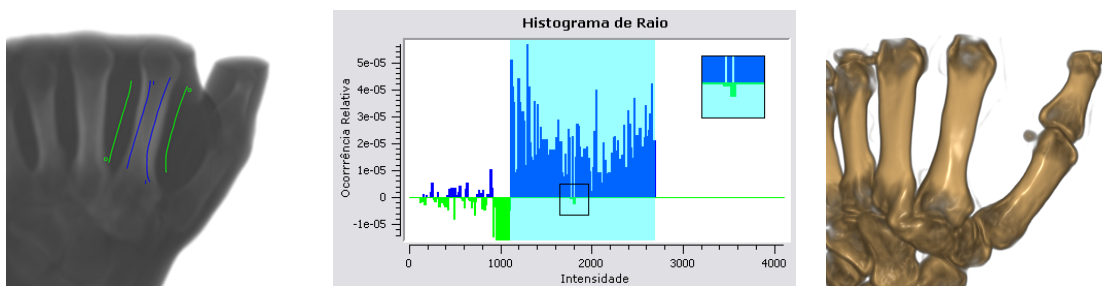


Figura 19 – Técnica de especificação de funções de transferência baseada em contorno aplicado no *dataset* de uma mão humana.

Fonte: Ropinski et al. (2008, p. 46, tradução nossa).

Mais recentemente, Guo, Mao e Yuan (2011) adaptaram o conceito WYSIWYG (do inglês, *What You See is What You Get*) à exploração volumétrica, implementando uma representação de alto nível semântico que disponibiliza ferramentas de desenho ao usuário.

3.3.3 Abordagens Dirigidas a Dados

De acordo com Kindlmann (2002), uma abordagem dirigida a dados adota basicamente duas estratégias distintas: assistir o usuário durante a especificação ou restringir o espaço exploratório, diminuindo a quantidade de graus de liberdade.

Um dos trabalhos precursores da categoria é o *Contour Spectrum* (BAJAJ; PASCUCCI; SCHIKORE, 1997). A técnica assimila recursos de Ajuste de Superfícies para descobrir isosuperfícies após uma especificação inicial (e.g., intensidade e magnitude de gradiente). Um pouco mais tarde, Kindlmann e Durkin (1998) também automatizaram a detecção do mesmo tipo de estrutura, mas por meio de histogramas de intensidade e gradientes de primeira e segunda ordem.

Sereda et al. (2006) utilizaram histogramas LH (do inglês, *low-high*) para facilitar a seleção de contornos em um volume. Em comparação a um histograma 2D convencional, como o apresentado na Figura 16, um histograma LH é caracterizado por conter perfis de intensidade, em vez de atributos adicionais – tipicamente magnitude de gradiente – associado aos *voxels*. A Figura 20 apresenta um histograma LH ajustado para enfatizar um cérebro no *dataset* de uma cabeça humana.

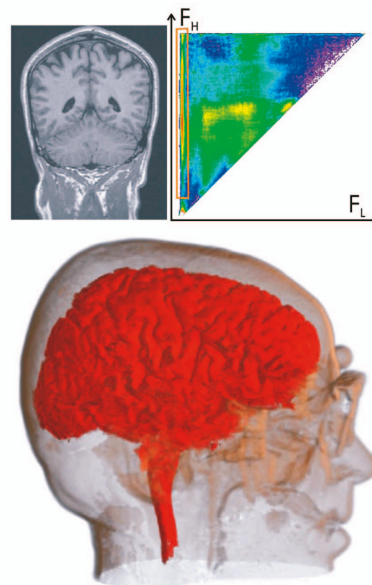


Figura 20 – Exploração volumétrica de uma cabeça humana em um histograma LH.

Fonte: Sereda et al. (2006, p.216).

A ideia de “visibilidade” também foi adaptada à interface de edição de FTs. Correa e Ma (2009) combinaram histogramas de visibilidade a uma abordagem não assistida. O usuário realiza uma especificação preliminar da FT, que é então refinada para tornar visíveis regiões e estruturas ocultas a partir da posição de observação. A Figura 21 apresenta um exemplo da execução da técnica no *dataset* de um pé humano. Na primeira configuração,

apenas a pele é visível (veja a Subfigura 21a), mas a estrutura óssea passa a ser revelada após a etapa de refinamento (veja a Subfigura 21b).

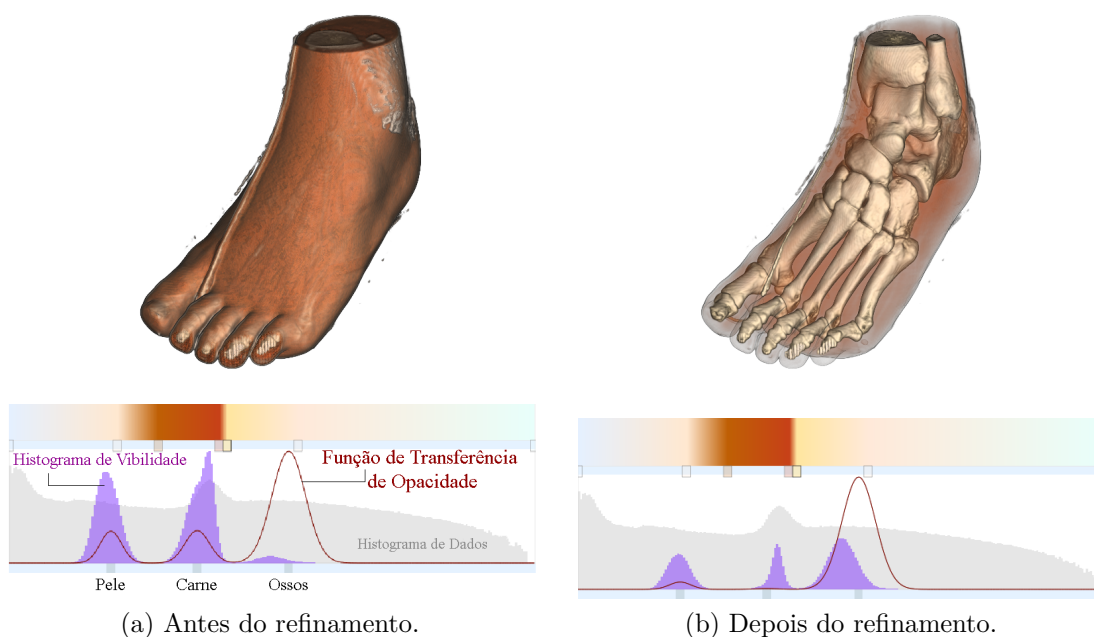


Figura 21 – Especificação de funções de transferência baseadas em visibilidade.

Fonte: Correa e Ma (2009, p.118, tradução nossa).

As técnicas apresentadas até aqui foram primordialmente propostas para manipular espaços de baixa dimensionalidade, geralmente com até três atributos. Todavia, de modo destoante ao que acontece em estratégias com imagens, abordagens dirigidas a dados não são facilmente extensíveis a funções de transferência de alta dimensionalidade (KINDLMANN, 2002). A partir deste momento, voltamos nossa atenção a abordagens planejadas para domínios de maior dimensionalidade.

Pradhan, Bartz e Mueller (2005) construíram uma interface de usuário de coordenadas paralelas (PCs, do inglês *Parallel Coordinates*). Cada atributo volumétrico é um eixo do gráfico e o valor especificado é um ponto de controle. Apesar de intuitivas para poucas dimensões, PCs são difíceis de ser interpretadas quando há um grande número de atributos envolvido. Zhao e Kaufman (2010) desenvolveram um sistema complementar a ideia anterior, que incorpora uma investigação global dos dados explorados por meio de um método de redução dimensional, o Mapeamento Topológico Localmente Linear (ROWEIS; SAUL, 2000), abreviado como LLE, do inglês *Locally Linear Embedding*. Guo, Xiao e Yuan (2011) utilizaram nuvens de dispersão geradas por um outro método de redução dimensional, a técnica de Escalonamento Multidimensional (KRUSKAL, 1964), MDS do inglês *Multidimensional Scaling*. Um exemplo da última abordagem é apresentado na Figura 22.

Ainda em relação à redução dimensional, Kim et al. (2010) desenvolveram um

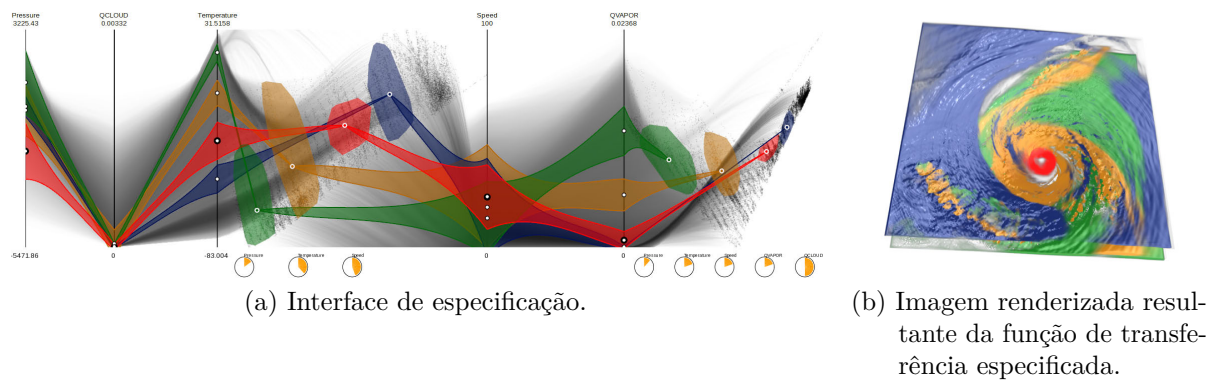


Figura 22 – Interface que combina coordenadas paralelas e gráficos de dispersão para especificação de uma função de transferência multidimensional em um volume multivariado.

Fonte: Guo, Xiao e Yuan (2011, p.19).

experimento para demonstrar que diferentes métodos podem ser combinados sem comprometer a visualização dos dados volumétricos. No experimento, domínios de dimensões arbitrárias são projetados em um espaço 2D e manipulados por meio de *widgets*. Entre os métodos de redução dimensional comparados estão: LLE (ROWEIS; SAUL, 2000), MDS (KRUSKAL; WISH, 1978), *Isomap* (TENENBAUM; SILVA; LANGFORD, 2000) e Análise de Componentes Principais (KPFERS, 1901).

Alguns autores têm empregado técnicas de agrupamento para automatizar a detecção de características presentes em um volume (LJUNG et al., 2016). Caban e Rheingans (2008) utilizaram o algoritmo de agrupamento *k-means* (STEINHAUS, 1956; MACQUEEN et al., 1967). Nguyen et al. (2012) aplicaram a técnica *Mean-Shift* (CHENG, 1995) e métodos de agrupamento hierárquicos sobre histogramas LH.

Pinto e Freitas (2007) optaram por reunir funções de base radial e Mapas Auto-Organizáveis (KOHONEN, 1990) – SOMs, do inglês *Self-Organizing Maps*. SOMs combinam redução dimensional e agrupamento, reduzindo dados de entrada para um espaço de menor dimensionalidade e ainda os organizando segundo uma relação de similaridade.

Recentemente, Cai et al. (2017) propuseram um sistema baseado em dois níveis de agrupamento que segue a ideia central do trabalho de Pinto e Freitas (2007). Em um primeiro momento, uma técnica de SOM é executada para produção de um mapa 2D. Em seguida, um segundo algoritmo de agrupamento, o corte normalizado, percorre o mapa inicial, formando células. A Figura 23 apresenta as imagens decorrentes da execução da técnica no *dataset* de um dente humano.

Um outro trabalho que também combina agrupamento e redução dimensional foi desenvolvido por Wang, Zhao e Kaufman (2012a). Os autores utilizaram árvores hierárquicas, os dendrogramas modificados. A Figura 24 apresenta um exemplo do dendrograma.

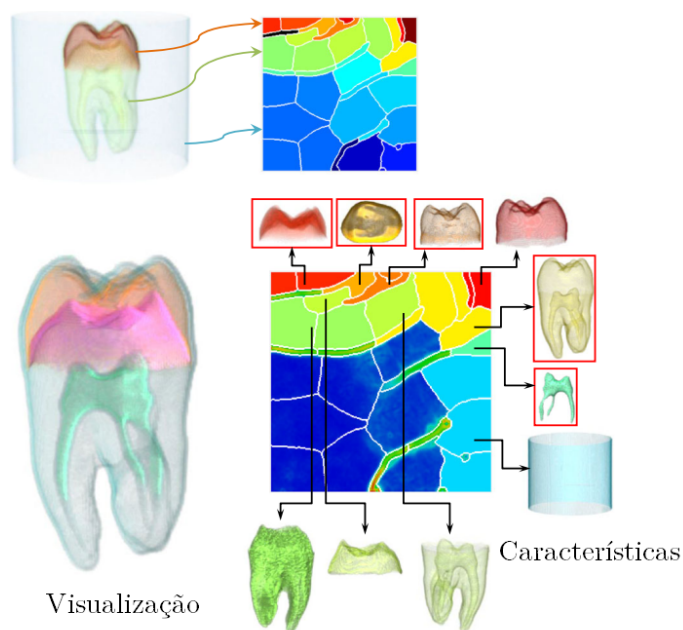


Figura 23 – Características identificadas em um dente humano. Atributos do domínio: intensidade, magnitude de gradiente, gradiente de segunda ordem, entropia, média, desvio padrão e assimetria.

Fonte: Cai et al. (2017, p.167, tradução nossa).

As barras representam aspectos volumétricos que foram identificados.

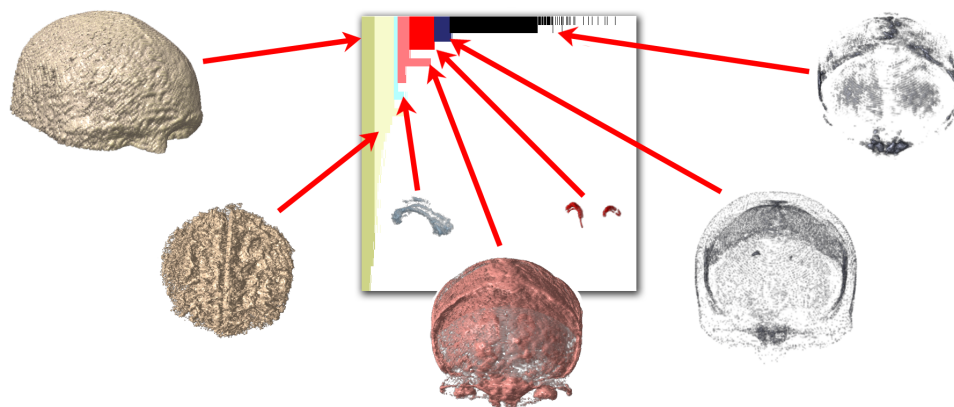


Figura 24 – Dendrograma modificado para a exploração do volume de uma cabeça humana.

Fonte: Wang, Zhao e Kaufman (2012a, p.122).

3.4 Considerações Finais

Durante o processo de exploração volumétrica, é essencial que funções de transferência que destaquem com precisão objetos volumétricos de um volume (LJUNG et al., 2016). Partindo desse princípio, é importante adicionar uma maior variedade de atributos

ao espaço da função e, assim, promover uma heterogeneidade nas assinaturas dos *voxels*. Contudo, é trabalhoso escolher atributos que garantam uma melhor classificação para cada *dataset*. [Arens e Domik \(2010\)](#) afirmam que não há um único tipo de atributo volumétrico adequado a todas as situações de exploração ([ARENS; DOMIK, 2010](#)).

Aquém da assinatura dos *voxels*, a dificuldade de se manipular parâmetros de especificação se alinha ao aumento de dimensionalidade e de complexidade dos *datasets* explorados ([ZHAO; KAUFMAN, 2010](#); [ZHOU; SCHOTT; HANSEN, 2012](#)).

4 Abordagem de Exploração de Volumes Multidimensionais

4.1 Considerações Iniciais

Nos Capítulos 2 e 3, elencamos perspectivas e problemas em aberto para o projeto e a especificação de funções de transferência. O cenário descrito se alinha aos desafios em aberto enunciados por [Ljung et al. \(2016\)](#). Toda a discussão pode ser condensada nos seguintes objetivos:

- manipular *datasets* maiores e dados volumétricos complexos com um baixo custo computacional;
- aumentar a efetividade de uma função de transferência ao classificar características volumétricas; e
- diminuir o esforço despendido para encontrar especificações que identifiquem objetos relevantes inseridos no volume.

A construção de ferramentas que permitam uma interação em tempo real é um dos principais desafios de visualização ([LACROUTE, 1995](#)) e tal fato não é diferente quando se trata de dados volumétricos. A experiência do usuário é diretamente afetada pelo tempo de execução. Tendo em vista que o processo de Visualização Direta de Volumes é, por si só, computacionalmente custoso – principalmente para a renderização de imagens de alta resolução e ricas em detalhes ([BEYER; HADWIGER; PFISTER, 2014](#)) – é oportuno desenvolver técnicas que sejam capazes de manipular espaços e *datasets* complexos, ao mesmo tempo que, não causem um sobrecarga em todo o esquema.

A habilidade de uma FT distinguir de modo apurado características depende da assinatura dos *voxels* e, invariavelmente, da utilização de domínios multidimensionais ([PINTO; FREITAS, 2007](#); [CAI et al., 2017](#)). Em detrimento a uma classificação mais robusta, o aumento da dimensionalidade torna a tarefa de especificação onerosa e menos intuitiva ([KINDLMANN; DURKIN, 1998](#); [LJUNG et al., 2016](#)). Ressalta-se que independentemente da combinação de atributos, a configuração de parâmetros durante a especificação é conjuntamente encarregada de propiciar a visualização de detalhes revelantes. Ambos os fatores envolvidos configuram um dilema.

Apesar de ser um campo de pesquisa bem ativo nas últimas duas décadas ([LJUNG et al., 2016](#)), a maior parte dos trabalhos acerca de FTs tende a se remeter a um desafio

isolado. A título de exemplo, há poucas abordagens para auxiliar a escolha de atributos para composição da assinatura dos *voxels* (ZHOU; SCHOTT; HANSEN, 2012).

Neste capítulo, apresentamos uma abordagem de exploração volumétrica que tem por objetivo mitigar os problemas discutidos. A abordagem compreende:

- uma heurística que auxilia a escolha de atributos para composição da assinatura dos *voxels*;
- um método semiautomático de classificação de características volumétricas;
- uma interface de edição baseada em um modelo de exploração constituído por um gráfico de dispersão, em que agrupamentos de pontos que representam características volumétricas.

Há secundariamente a geração automática de uma especificação da FT com base nos resultados produzidos pelo método de classificação de características.

A abordagem simplifica o processo de especificação de FTs multidimensionais e a exploração volumétrica, reduzindo drasticamente a quantidade de variáveis que podem ser ajustadas em comparação à edição manual. Um esquema que ilustra a interação na abordagem é apresentado na Figura 25.

O Passo 1 estabelece um *ranking* de candidatos que funciona como um guia para seleção de atributos volumétricos. A decisão efetiva pela combinação de atributos permanece sobre responsabilidade do usuário. O *ranking* apenas indica a ordem em que os atributos devem ser escolhidos com base em uma pontuação calculada por operadores estatísticos.

O usuário deve no Passo 2 informar parâmetros para a descoberta de características volumétricas e a construção de um gráfico de dispersão, no constituído por agrupamentos. No gráfico, os *voxels* são representados por pontos. Cada agrupamento é um material, estrutura ou qualquer outra característica identificada pelo método de classificação. A precisão dos agrupamentos produzidos depende dos parâmetros informados e dos atributos volumétricos definidos no Passo 1. Diante disso, é em algumas vezes inevitável ter que refinar os parâmetros ou retroceder ao passo anterior para modificar a assinatura definida. O procedimento é realizado de forma empírica.

Por fim, o Passo 3 garante a exploração do volume mediante a navegação pelo gráfico de dispersão de características, selecionando pontos ou agrupamentos. A cada conjunto selecionado, os *voxels* associados à região correspondente são destacados e a configuração resultante é renderizada.

Nas próximas seções, cada componente da abordagem proposta é descrito em maiores detalhes. Antes disso, ressalta-se que apesar da organização do texto, a construção

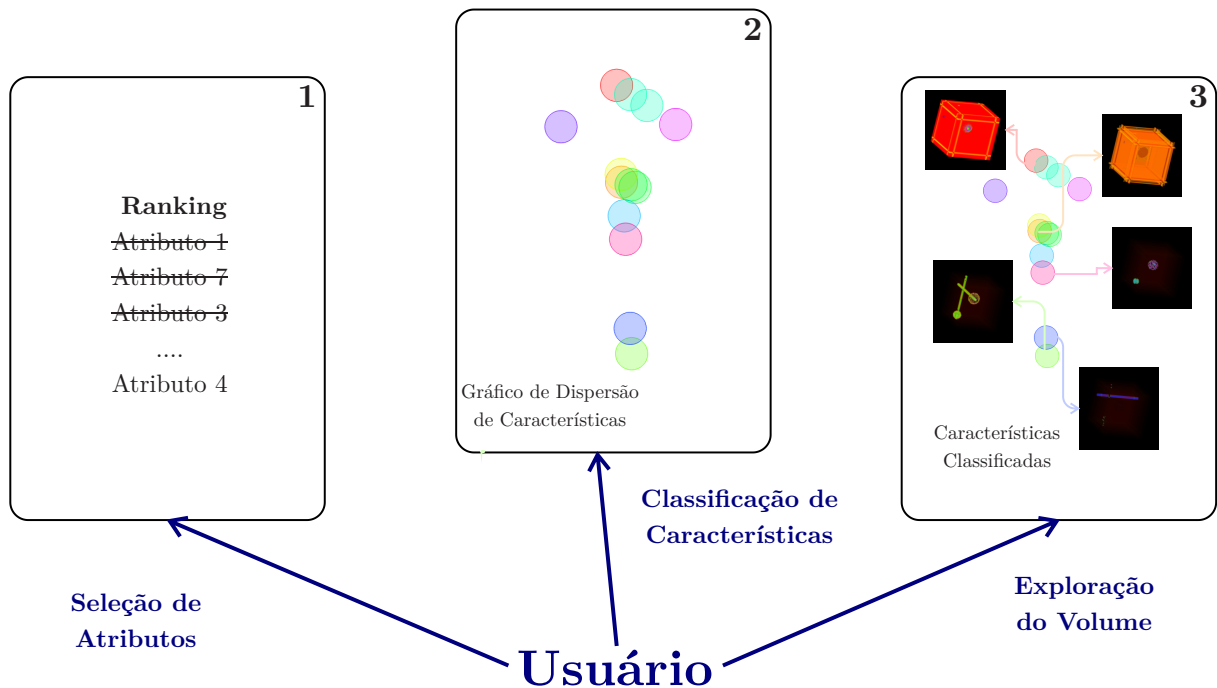


Figura 25 – Exemplo de interação do usuário para exploração do volume por meio da abordagem proposta.

do gráfico de dispersão de características e integralmente da interface de navegação é atrelada à execução do método de classificação.

4.2 Seleção de Atributos Volumétricos

A pura adição de atributos volumétricos nem sempre melhora um classificador (ZHOU; SCHOTT; HANSEN, 2012). Em outras palavras, aumentar a dimensionalidade de uma FT não é garantia de identificação mais acurada de objetos contidos no volume. Segundo Indyk e Motwani (1998), esse problema é, em diferentes áreas da ciência, chamado de Maldição da Dimensionalidade (do inglês, *Curse of Dimensionality*). Algumas estratégias para suprimir tal efeito foram propostas na literatura (GUYON; ELISSEEFF, 2003). Como os dados não são rotulados, é necessário adotar técnicas não supervisionadas para, a partir de um conjunto de atributos disponíveis encontrar, encontrar uma configuração que forneça uma classificação mais apurada (DY; BRODLEY, 2004). Estratégias do tipo são pouco exploradas e, assim, um desafio em aberto na literatura (QIAN; ZHAI, 2013). Tratando-se de FTs, a adoção de abordagens inseridas nesse contexto é ainda mais incipiente. Uma das poucas abordagens propostas nesse sentido é o trabalho de Zhou, Schott e Hansen (2012).

Zhou, Schott e Hansen (2012) introduziram uma heurística que, dado um conjunto V de n atributos volumétricos, indica o par de atributos que combinados produzam uma identificação mais efetiva das características. Para isso, são empregadas duas pro-

priedades estatísticas recorrentes na área de seleção de características, a entropia e a correlação (FRENKEN et al., 2007).

O algoritmo pode ser decomposto em dois procedimentos de busca, um para cada elemento da combinação. O primeiro procedimento é uma instrução simples: optar pela intensidade ou pelo atributo com maior entropia. Em contrapartida, para seleção do segundo elemento, o algoritmo atua sobre uma pontuação composta pela correlação com atributo inicial, além da entropia do próprio atributo buscado. O cálculo da pontuação é definido na Equação 4.1.

$$s_i = -\alpha|Corr(Y_j, Y_i)| + (1 - \alpha)\frac{E(Y_i)}{\max E(Y)}, 0 \leq \alpha \leq 1, \quad (4.1)$$

onde s_i é a pontuação, Y_i é o atributo analisado, Y_j é o atributo inicial, $Corr$ é a função de correlação e E é a de entropia e α , o fator de decisão. A ideia por trás dos procedimentos consiste em escolher o atributo com maior entropia mas, ao mesmo tempo, com menor correlação com o atributo inicial. No cálculo, o valor da entropia é normalizado. O fator α pode ser variado no intervalo $[0, 1]$ para permitir ao usuário regular qual propriedade terá contribuição mais preponderante na pontuação. Um valor mais próximo de 0 favorece a correlação e proporcionalmente um valor mais próximo de 1 favorece a entropia.

O conceito de entropia pode ser interpretado como a quantidade de informação presente no sistema proposto por Frenken et al. (2007). Considerando o caso discreto, uma entropia E pode ser calculada como:

$$E(x) = -\sum_{i=1}^n p(X_j)\log_b p(x), \quad (4.2)$$

onde b é uma constante $\in \mathbb{R}$, p_x é a probabilidade de x acontecer e n , a quantidade de classes do histograma. Para $b = 2$, o operador é conhecido como Entropia de Shannon (1948).

Coefficiente de correlação é uma medida estatística que expressa o grau de relacionamento (dependência) entre duas variáveis aleatórias, sem interferência dos próprios valores assumidos pelas variáveis (BODDY; SMITH, 2009). O cálculo do coeficiente de correlação é definido em função da covariância e, a grosso modo, pode ser interpretado como uma “normalização” de tal operador. A Equação 4.3 descreve formalmente o cálculo, considerando o clássico operador de Pearson (PEARSON, 1901).

$$CorrY_i, Y_j = \frac{Cov(Y_i, X_j)}{\sqrt{Cov(Y_i, Y_i)Cov(Y_j, Y_j)}}, \quad (4.3)$$

onde $Corr$ é o coeficiente de correlação e Cov , o operador de covariância.

Embora argumentem em favor da utilização de espaços bidimensionais devido ao processo de especificação, Zhou, Schott e Hansen (2012) afirmam que a utilização de

mais do que dois atributos volumétricos geralmente proporciona uma classificação mais sofisticada. Em consonância a isso, adaptamos a técnica original a fim de gerar um *ranking* de candidatos, ou seja, uma ordem de atributos assinalados para compor a assinatura dos *voxels*. A heurística modificada é apresentada no Algoritmo 1. A principal diferença da versão original se resume à utilização de correlação múltipla.

Algoritmo 1: Heurística para geração do *ranking* de atributos candidatos.

	Entrada: $\mathbb{A}$	/* Atributos volumétricos */
	Saída: $\mathbb{R}$	/* Ranking de Atributos */
1	início	
2	$\mathbb{R} \leftarrow \{\}$	/* Atributos escolhidos */
3	$\mathbb{D} \leftarrow \mathbb{A}$	/* Atributos disponíveis */
4	$d \leftarrow \max(\text{Entropia}(\mathbb{D}))$	
5	$\mathbb{D} \leftarrow \mathbb{D} - \{d\}$	
6	enquanto $\mathbb{D} \neq \{\}$ faça	
7	$d \leftarrow \max(\text{Pontuação}(\mathbb{D}, \mathbb{R}))$	
8	$\mathbb{D} \leftarrow \mathbb{D} - \{d\}$ $\mathbb{R} \leftarrow \mathbb{R} + \{d\}$	
9	fim	
10	fim	

Os atributos são organizados em dois conjuntos: disponíveis e escolhidos. Inicialmente, todos os atributos estão no primeiro conjunto. O atributo de maior entropia é então removido para ser adicionado ao conjunto de escolhidos. Esse atributo é, assim, primeiro do *ranking*. Nas próximas iterações a pontuação computada pela Equação 4.1 passa a definir o atributo que deve ser removido do conjunto de disponíveis. É importante observar que o cálculo descrito na Equação 4.3 não pode mais ser adotado. A partir de dois atributos escolhidos, deve ser utilizada correlação múltipla. Existem diferentes formas de se calcular correlação múltipla, porém neste trabalho optamos por utilizar regressão linear como definido por [Abdi \(2007\)](#).

A partir do *ranking*, o usuário deve empiricamente decidir quais atributos escolher para compor a assinatura dos *voxels*. Veja que a ideia-chave desse *ranking* é que sejam antepostos os n melhores candidatos com base em uma ordem por pontuação. Entretanto, o número n – i.e, a quantidade mínima de candidatos para se obter uma classificação de características relevantes – é de responsabilidade do usuário. Independentemente da ordem de tais atributos, o usuário tem total liberdade para estabelecer qualquer combinação, seja se baseando em um conhecimento prévio sobre os dados explorados ou de forma experimental. Cabe ressaltar que a adição desmedida de atributos pode ocasionar uma elevada sensibilidade a falso-positivos. Fortuitamente, essa adição pode fragmentar os agrupamentos de modo a tornar a exploração do gráfico exaustiva.

Os procedimentos executados durante a regressão linear para um *dataset* sufici-

entamente grande tem a complexidade de tempo de $O(c^2n)$, onde c é a dimensão e n o número de instâncias. Como a heurística executa c regressões lineares, o algoritmo tem complexidade total de $O(c^3n)$.

4.3 Método de Classificação de Características Volumétricas

O método de classificação de características é retratado na Figura 26. O método combina técnicas de redução dimensional, agrupamento e seleção de representativos.

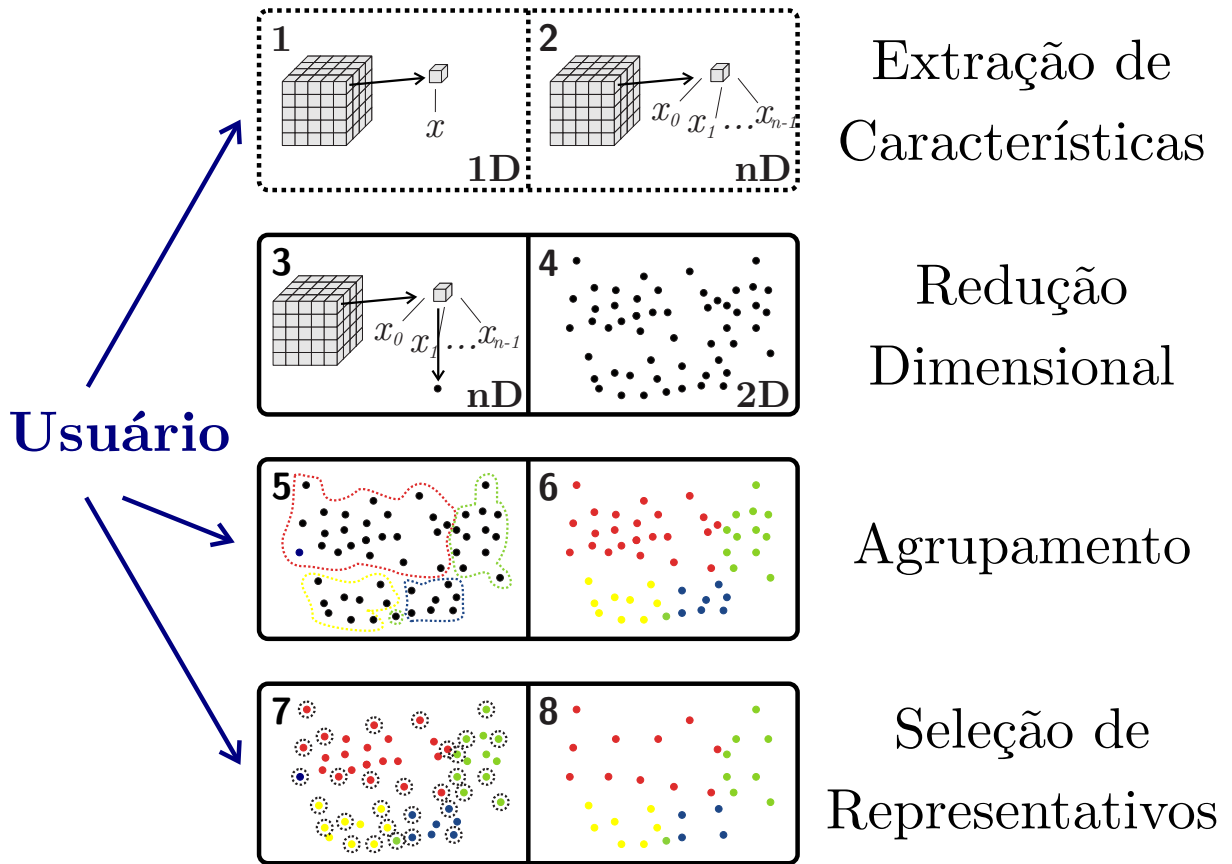


Figura 26 – Visão geral das atividades compreendidas no método de classificação de características volumétricas.

Em uma etapa preliminar (Quadros 1 e 2), o usuário escolhe operadores para extração de dados adicionais com base no *ranking* de candidatos fornecido pela etapa de seleção de atributos. Dados originalmente unidimensionais (1D) são derivados para formação de um espaço multidimensional (nD). Após a obtenção de atributos adicionais, os dados são organizados em uma grade regular e o método de classificação de características é realmente iniciado. Na primeira etapa (Quadros 3 e 4), uma técnica de redução dimensional transforma todos os *voxels* em pontos 2D.

A segunda etapa (Quadros 5 e 6) executa um processo de agrupamento sobre a nuvem de pontos decorrente da etapa anterior. Como resultado, cada agrupamento é

presumivelmente uma característica volumétrica diferente.

Na última etapa (Quadros 7 e 8), uma técnica de seleção de representativos simplifica a nuvem de pontos, diminuindo a densidade visual de cada agrupamento. Uma discussão mais detalhada a respeito do gráfico de dispersão e da interface implementada é apresentada na Seção 4.4.

A seguir, descrevemos cada uma das etapas do método de classificação.

4.3.1 Redução Dimensional

Quando é necessário analisar dados multidimensionais é quase imediato pensar em redução dimensional. Todavia, é trabalhoso escolher uma estratégia apropriada a cada *dataset* explorado. Mesmo com escopo limitado à especificação de FTs, uma grande variedade de técnicas já foi utilizada, tais como: Isomap (TENENBAUM; SILVA; LANGFORD, 2000), MDS (KRUSKAL; WISH, 1978), LLE (ROWEIS; SAUL, 2000) e PCA (KPFERS, 1901). Ao ponderar que a etapa (veja os Quadros 4 e 5) posterior à redução dimensional envolve um processo de agrupamento, o Escalonamento Multidimensional (MDS, do inglês *Multidimensional Scaling*) desponta como uma importante alternativa. O MDS tem como principal vantagem preservar ao máximo as relações de similaridades originais entre os elementos de um conjunto (GHODSI, 2006). Diferentes autores – entre os quais Luo, Wilson e Hancock (2003), Heesch, Yavlinsky e Rüger (2006) e Moya-Anegón, Herrero-Solana e Jiménez-Contreras (2006) – já demonstraram que técnicas de agrupamento podem ser executadas de modo subsequente ao MDS e ocasionalmente produzir resultados precisos.

Ainda que mostre um rumo, o MDS não resolve de imediato o problema. O algoritmo tem a complexidade de tempo de $O(n^3)$ e é, portanto, quase proibitivo adotá-lo para uma grande quantidade de instâncias (HOAND, 2008). Apesar disso, há algumas técnicas derivadas que têm uma menor complexidade de tempo e que podem ser perfeitamente empregadas em substituição à versão original do algoritmo (PLATT, 2005). Optou-se por utilizar uma dessas técnicas, o *FastMap* (FALOUTSOS; LIN, 1995).

FastMap é uma variante do MDS, introduzida por Faloutsos e Lin (1995), tipicamente utilizada para projetar dados em espaços de baixa dimensionalidade, principalmente euclidianos (2D ou 3D). A técnica tem como vantagens: uma elevada acurácia com o MDS e uma complexidade linear, na verdade $O(nk)$ onde k é a dimensão alvo (PAULOVICH; SILVA; NONATO, 2010). Ambos os fatores justificam a escolha do *FastMap* em face da necessidade de se manipular *datasets* grandes sem prejuízo aos recursos computacionais e à separabilidade dos agrupamentos. Melhor dizendo, o *FastMap* alia o baixo custo computacional à capacidade de preservar distâncias entre objetos (FODOR, 2002).

Considerando um *dataset* inicial de tamanho n e dimensão d , o *FastMap* projeta n instâncias em um espaço k -dimensional, no qual $k < d$. Neste trabalho, n é o número

total de *voxels*, d é a dimensão do domínio e $k = 2$. O algoritmo é recursivo e pode ser sucintamente descrito nos seguintes passos:

1. encontrar dois pivôs de um *dataset*, i.e., os dois pontos mais afastados entre si;
2. projetar os demais pontos em um hiperplano ortogonalmente posicionado entre os pivôs.

As estratégias capazes de assinalar precisamente pivôs têm a complexidade no mínimo quadrática. Como a adoção dos métodos comprometeria o tempo de execução, Faloutsos e Lin (1995) propuseram a heurística que é apresentada no Algoritmo 2.

Algoritmo 2: Escolha de pivôs do *Fastmap*.

Entrada:	$\mathbb{P}$	/* <i>Dataset</i> */
Saída:	O_a, O_b	/* Pivôs */
1	início	
2	$O_a \leftarrow$ ponto arbitrário $p \in \mathbb{P}$	/* Pivô O_a temporário */
3	$O_b \leftarrow$ ponto $p \in \mathbb{P}$ mais distante de O_a	/* Pivô O_b */
4	$O_a \leftarrow$ ponto $p \in \mathbb{P}$ mais distante de O_b	/* Pivô O_a */
5	fim	

As projeções dos pontos restantes são computadas por meio da Lei dos Cossenos. A Figura 27 apresenta uma ilustração da projeção de um ponto p em um hiperplano ortogonal ao segmento de reta formado pelos pivôs O_a e O_b .

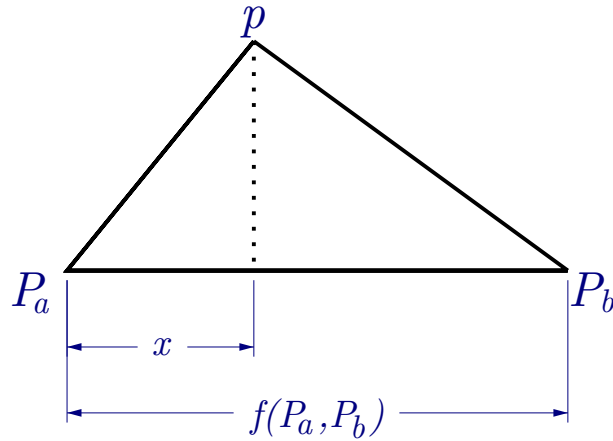


Figura 27 – Lei dos Cossenos e Teorema de Pitágoras para projeção de um ponto p no hiperplano ortogonal ao segmento de reta $\overline{O_a O_b}$.

Fonte: adaptado de Faloutsos e Lin (1995, p.169).

A operação de projeção apresentada na Figura 27, pode ser algebricamente expressa como:

$$f(O_b, p)^2 = f(O_a, p)^2 + f(O_a, O_b)^2 - 2xf(O_a, O_b), \quad (4.4)$$

onde p é o ponto original, x é a projeção computada, O_a e O_b são os pivôs e f , a função de distância.

Reescrevendo a Equação 4.4 para colocar x em evidência, temos:

$$x = \frac{f(O_a, p)^2 + f(O_a, O_b)^2 - f(O_b, p)^2}{2f(O_a, O_b)} \quad (4.5)$$

A distância entre dois pontos p_a e p_b para a primeira dimensão ($i = 1$) depende apenas das posições originais, mas para as demais dimensões, as coordenadas já projetadas passam a ser inseridas no cálculo. Considerando que $dist$ é um operador de distância métrica (e.g., distância euclidiana) e i é o número de dimensões computadas, a equação que calcula f para $i \geq 2$ é definida como:

$$f(p_a, p_b) = \sqrt{\text{dist}_d(p_a, p_b)^2 - \text{dist}_{i-1}(\text{proj}_a, \text{proj}_b)^2}, \quad (4.6)$$

onde proj_a e proj_b são respectivamente as projeções dos pontos p_a e p_b em $i - 1$, e d é a dimensão original.

4.3.2 Agrupamento

Agrupamento é um ramo do aprendizado não supervisionado que tem por objetivo encontrar elementos similares entre si. Assim como em redução dimensional, não há diretrizes universais que evitem uma escolha *ad hoc* das técnicas (JAIN; MURTY; FLYNN, 1999). Um fator também agravante é que dados volumétricos podem ser oriundos de diferentes fontes, o que privilegia uma maior imprevisibilidade. Diante de tudo isso, optou-se por uma técnica de agrupamento baseada em densidade bem conhecida por sua computação eficiente e pela pouca sensibilidade à natureza dos dados (GAN; TAO, 2015), o DBSCAN (ESTER et al., 1996).

O *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN) é um algoritmo de agrupamento proposto por Ester et al. (1996) que se fundamenta na execução de *range-queries* (consultas por abrangência). Em comparação às demais técnicas de agrupamento, o DBSCAN tem por vantagens: a capacidade de identificar regiões de formato arbitrário e a baixa suscetibilidade a heterogeneidade dos artefatos presentes nos dados (XU; TIAN, 2015). O algoritmo atua sobre dois parâmetros de entrada: o raio de busca ε (*eps*) e o número mínimo de pontos *minPts*. Assumindo $\mathbb{P}$ como um *dataset* de pontos, o modelo de agrupamento do DBSCAN atende as seguintes definições:

Definição 1. Um ponto $p \in \mathbb{P}$ pode ser rotulado como: *core-point* (ponto-central), *border-point* (ponto de borda) ou *noise* (ruído).

Definição 2. Um ponto $p_a \in \mathbb{P}$ é alcançável por um ponto $p_b \in \mathbb{P}$ se $\text{dist}(p_a, p_b) \leq \varepsilon$.

Definição 3. A vizinhança de um ponto $p \in \mathbb{P}$, denotada por $V_\varepsilon(p)$, é obtida pela execução de uma range-query de raio ε em $\mathbb{P}$, tendo como ponto de partida o próprio ponto p .

Definição 4. Um ponto $p \in \mathbb{P}$ é rotulado como *core-point* se $|V_\varepsilon(p)| \geq \text{minPts}$;

Definição 5. Um ponto $p_a \in \mathbb{P}$ é *border-point* se alcançável por um *core-point* $p_b \in \mathbb{P}$, mas $|V_\varepsilon(p_a)| < \text{minPts}$.

Definição 6. Pontos $p_i \in \mathbb{P}$ que não atendam as Definições 4 e 5 são rotulados como *noise*.

Definição 7. Um agrupamento $\mathbb{A}$ é um subconjunto não vazio de $\mathbb{P}$.

Definição 8. $\forall p_i \in \mathbb{A}$ e $|\mathbb{A}| > 1$, $\exists p_j \in \mathbb{A}$, tal que p_i é alcançável por p_j e $p_i \neq p_j$.

Um exemplo da execução do algoritmo é apresentado na Figura 28. As setas indicam os pontos mutuamente alcançáveis e as circunferências delimitam o raio ε de consulta. Para $\text{minPts} = 3$, todos os pontos em vermelho (e.g., ponto A) são *core-points*, os pontos em amarelo (pontos B e C) são *border-points* e o ponto em azul (ponto N) é rotulado como *noise*. Tanto os *core-points* quanto os *border-points* pertencem ao mesmo agrupamento.

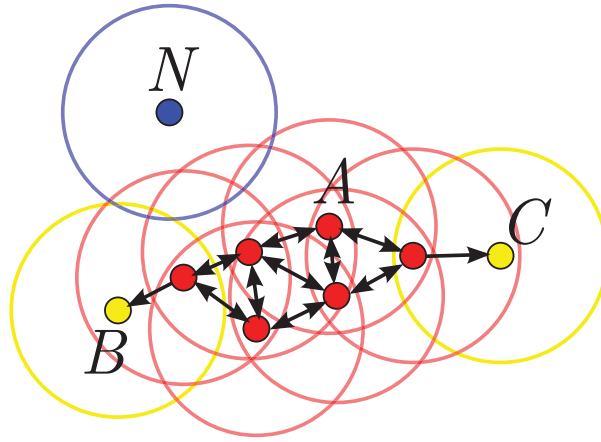


Figura 28 – Exemplo de execução do DBSCAN.

Fonte: Schubert et al. (2017, p. 19:3).

O DBSCAN é organizado em duas rotinas básicas: uma rotina principal de criação de agrupamentos e uma secundária que é responsável pelo processo de expansão. A rotina principal é detalhada no Algoritmo 3. Todos os pontos são inicialmente rotulados como não classificados. A cada iteração, o algoritmo escolhe um *core-point* p ainda não visitado como ponto inicial de um agrupamento em expansão.

A rotina de expansão é apresentada no Algoritmo 4. A efetiva instauração do processo depende do resultado de uma *range-query* com raio de busca ε centrada no *core-point* p . A expansão não ocorre se o número de pontos recuperados for inferior a

Algoritmo 3: Rotina principal do DBSCAN.

Entrada: $\mathbb{P}$, ε , minPts

```

/*  $\mathbb{P}$  Dataset */
/*  $\varepsilon$  Distância máxima */
/*  $\text{minPts}$  Número mínimo de pontos */
/*  $\text{clusterID}$  Índice do agrupamento em expansão */
/* UNCLASSIFIED Ponto não classificado */
/* NOISE Ponto que não pertence a nenhum agrupamento */

1 início
2   clusterID  $\leftarrow$  0      /* Inicialização do contador de agrupamentos */
3   para todo  $p \in \mathbb{P}$  faça
4     Rotular  $p$  como UNCLASSIFIED
5   fim
6   para todo  $p \in \mathbb{P}$  faça
7     se  $p$  é UNCLASSIFIED então
8       se EXPANDCLUSTER( $p$ ,  $\varepsilon$ ,  $\text{minPts}$ ,  $\mathbb{P}$ , clusterID) então
9         clusterID  $\leftarrow$  clusterID + 1      /* sucesso, tentar novo
          agrupamento */
10      fim
11    fim
12  fim
13 fim

```

minPts . Nesse caso, o ponto p é rotulado como *noise* e a execução do DBSCAN retorna ao Algoritmo 3.

Uma vez iniciado o processo de expansão, todos os pontos recuperados ainda não classificados – com exceção do próprio ponto inicial – são adicionados a uma lista de pontos expansíveis S . Nas iterações que se sucedem, o processo continua até que a lista S esteja vazia. Cada ponto alcançado que não pertencer a um agrupamento, independentemente se não classificado ou *noise*, é atribuído ao agrupamento atualmente em expansão.

Quando o DBSCAN termina, cada agrupamento identificado é uma característica do volume. Além do mais, neste momento cada *voxel* é representado por um ponto.

4.3.3 DBSCAN 2D

Segundo Gan e Tao (2015), a complexidade de tempo do DBSCAN é, em uma análise breve, diretamente proporcional ao custo unitário de uma *range-query*. Ao todo, n operações do tipo são realizadas durante a execução do algoritmo. Ao analisar a relação instâncias/*range-queries*, muitos pesquisadores – inclusive os próprios autores, Ester et al. (1996) – sempre anunciaram que estruturas de indexação (e.g., *VP-trees* e *KD-trees*) poderiam diminuir o tempo de execução para propiciar uma complexidade $O(n \log n)$.

Algoritmo 4: Função EXPANDCLUSTER.

```

/*  $p$  Ponto central                                     */
/*  $\varepsilon$  Distância máxima                             */
/*  $minPts$  Número mínimo de pontos                       */
/*  $clusterID$  Índice do agrupamento em expansão        */
/* UNCLASSIFIED Ponto não classificado                  */
/* NOISE Ponto que não pertence a nenhum agrupamento   */
1 função EXPANDCLUSTER( $p, \varepsilon, minPts, \mathbb{P}, clusterID$ ): booleano
2    $N \leftarrow \text{RANGEQUERY}(p, \varepsilon, \mathbb{P})$ 
3   se  $|N| < minPts$  então
4     Rotular  $p$  como NOISE
5     retorna false
6   senão
7     Atribuir  $p$  ao agrupamento  $clusterID$ 
8     para todo  $n \in N$  faça
9       Atribuir  $n$  ao agrupamento  $clusterID$ 
10    fim
11     $S \leftarrow S - \{p\}$ 
12    para todo  $s \in S$  faça
13       $N \leftarrow \text{RANGEQUERY}(s, \varepsilon, \mathbb{P})$ 
14      se  $|N| \geq minPts$  então
15        Atribuir  $s$  ao agrupamento  $clusterID$ 
16        para todo  $n \in N$  faça
17          se  $n$  é UNCLASSIFIED então
18             $S \leftarrow S \cup \{n\}$ 
19            Atribuir  $n$  ao agrupamento  $clusterID$ 
20          fim
21          se  $n$  é NOISE então
22            Atribuir  $n$  ao agrupamento  $clusterID$ 
23          fim
24        fim
25      fim
26    fim
27    retorna true
28  fim
29 fim

```

No entanto, algumas pesquisas recentes – como é o caso dos trabalhos de [Gunawan e Berg \(2013\)](#), [Gan e Tao \(2015\)](#) e [Schubert et al. \(2017\)](#) – demonstraram que a última afirmação é falha. A maioria das estratégias que garantem um baixo tempo de execução geram resultados discrepantes dos que são organizados pelo algoritmo original. Na verdade, é possível construir uma versão do DBSCAN com complexidade de $O(n \log n)$ para espaços 2D. Contudo, para mais do que duas dimensões ($d \geq 3$), a complexidade mínima

é teoricamente de $\Omega(n^{\frac{4}{3}})$, como foi apontado em revisões de Gunawan e Berg (2013), Gan e Tao (2015) e Schubert et al. (2017). Na prática, a complexidade pode ainda ser de $O(n^2)$ para situações que não são feitas escolhas adequadas dos parâmetros.

Gunawan e Berg (2013) propuseram uma versão do DBSCAN que resolve de forma acurada o problema original para espaços 2D em uma complexidade de tempo $O(n \log n)$. A versão foi implementada neste trabalho.

A técnica de Gunawan e Berg (2013) tem por base o mapeamento do *dataset* para uma grade 2D, onde cada célula é um quadrado de lado $\varepsilon/\sqrt{2}$. Além da grade, grafos e *spanning-trees* (árvores de extensão) são empregados para acelerar o processamento. O algoritmo atende as definições da versão original e se organiza em quatro etapas:

1. construção da grade 2D;
2. determinação de *core-points*;
3. combinação de agrupamentos;
4. determinação de *border-points* e *noise*.

Na primeira etapa, todos os pontos do *dataset* são analisados para demarcar as coordenadas limítrofes e, conseqüentemente, a dimensão mínima da grade. Veja que essa grade é um retângulo regular formada por células quadradas, nas quais os pontos devem estar totalmente inseridos.

Gunawan e Berg (2013) recomendam a utilização de uma estrutura esparsa (e.g., tabela *hash*) para armazenar a grade, pois, a maior parte das células será vazia. Assumindo essa estratégia, o processo de construção é apresentado no Algoritmo 5.

Assim que a grade é gerada, as células não vazias devem ser examinadas para rotular os pontos. A principal propriedade associada a grade é estabelecida na Definição 9. A propriedade permite aferir que dois pontos inseridos em uma mesma célula distam até ε um do outro.

Definição 9. $\forall$ pontos $p_a, p_b \in$ célula c , $dist(p_a, p_b) \leq \varepsilon$.

A segunda etapa é responsável por identificar *core-points*. O algoritmo analisa duas situações distintas. Primeiro, se a quantidade de pontos inseridos em uma célula for maior ou igual a *minPts*, todos os pontos são rotulados como *core-points* com base na Definição 9. Segundo, caso haja menos do que *minPts* pontos na célula, uma contagem por meio de força bruta estipula o número de pontos alcançáveis pelo ponto a ser rotulado. A região de vizinhança que deve ser verificada nesse último caso é apresentada na Figura 29. O procedimento é apresentado no Algoritmo 6.

Algoritmo 5: Construção da grade 2D.

Entrada: $\mathbb{P}, \varepsilon$
Saída: G

```

/*  $G$  grade de células de lado  $\varepsilon/\sqrt{2}$  */
/*  $\mathbb{P}$  dataset */
/*  $nRows$  Número de linhas */
/*  $nCols$  Número de colunas */
1 início
2   Inicializar  $minX, maxX, minY$  e  $maxY$ 
3    $cellSide \leftarrow \frac{\varepsilon}{\sqrt{2}}$  /* Tamanho da célula */
4    $nRows \leftarrow \lfloor \frac{maxX - minX}{cellSide} + 1 \rfloor$ 
5    $nCols \leftarrow \lfloor \frac{maxY - minY}{cellSide} + 1 \rfloor$ 
6   Inicializar  $G$  como uma grade vazia de dimensão  $nRows \times nCols$ 
7   para cada ponto  $p \in \mathbb{P}$  faça
8     | Adicionar o ponto  $p$  à célula  $G \left[ \lfloor \frac{x - minX}{cellSide} + 1 \rfloor \right] \left[ \lfloor \frac{y - minY}{cellSide} + 1 \rfloor \right]$ 
9   fim
10 fim

```

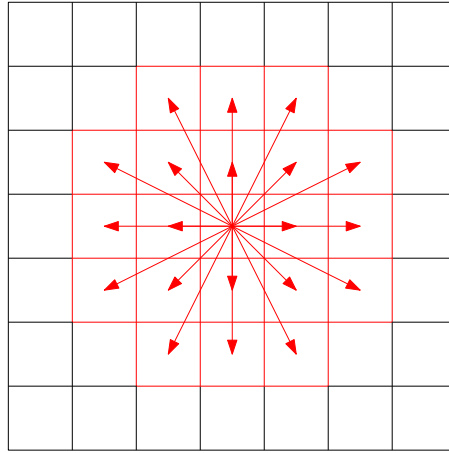


Figura 29 – Vizinhança de uma célula central.

Fonte: Gunawan e Berg (2013, p. 15).

A terceira etapa inicia o processo de identificação dos agrupamentos. Todos os *core-points* são atribuídos aos seus respectivos agrupamentos. A etapa assume que as células da grade são pontos de um grafo para execução do algoritmo de componentes conexas (TARJAN, 1972). O grafo e o algoritmo de componentes conexas consideram que:

Definição 10. Uma célula c é um ponto se $\exists p \in c$, tal que p é rotulado como *core-point*.

Algoritmo 6: Determinação de *core-points*.**Entrada:** $\mathbb{P}$, ε , minPts , G

```

1  início
    /*  $G$  grade de células de lado  $\varepsilon/\sqrt{2}$  */
    /*  $\mathbb{P}$  dataset */
2  para cada célula  $c$  não vazia  $\in G$  faça
3      se  $|c| > \text{minPts}$  então
4          para todo ponto  $p \in c$  faça
5              Rotular o ponto  $p$  como core-point
6          fim
7      senão
8          para cada  $p \in c$  faça
9              nPoints  $\leftarrow 0$ 
10             para cada célula  $nc \in \text{vizinhança } V_\varepsilon(c)$  faça
11                 para cada  $q \in nc$  faça
12                     se  $\text{dist}(p, q) \leq \varepsilon$  então
13                         nPoints  $\leftarrow \text{nPoints} + 1$ 
14                         se nPoints  $\geq \text{minPts}$  então
15                             Rotular o ponto  $p$  como core-point
16                             break
17                     fim
18                 fim
19             fim
20             se nPoints  $\geq \text{minPts}$  então
21                 break
22             fim
23         fim
24     fim
25 fim
26 fim
27 fim

```

Definição 11. Uma célula c_i é conectada a uma célula c_j se $\text{dist}(p_a, p_b) \leq \varepsilon$, onde p_a *core-point* $\in c_i$ e p_b *core-point* $\in c_j$.

Definição 12. Toda componente conexa é um agrupamento.

A vizinhança que deve ser verificada na Definição 11 é a mesma que a apresentada na Figura 29. Gunawan e Berg (2013) propõe a utilização do Diagrama de Voronoi ou de uma abordagem de força bruta, mas árvores de indexação métrica também podem ser utilizadas.

A última etapa do algoritmo determina quais pontos são *border-points* ou *noises* e também atribui cada *border-point* ao agrupamento correto. Todos os passos executados nessa etapa são enunciados no Algoritmo 7. Células com quantidade de pontos $\geq \text{minPts}$

não são checadas. Os pontos ainda não rotulados são atribuídos ao agrupamento do *core-point* mais próximo se a distância for de até ε . Caso não haja nenhum ponto até a distância estipulada, p_a é rotulado como *noise*.

Algoritmo 7: Determinação de *border-points* e *noises*.

Entrada: $\mathbb{P}$, ε , G

/ G grade de células de lado $\varepsilon/\sqrt{2}$*

**/*

```

1 início
2   para cada célula  $c$  (não vazia)  $\in G$  faça
3     para cada ponto  $p \in$  célula  $c$  faça
4       se  $p$  não é core-point então
5         nearestPoint  $\leftarrow$  NULL
6         para cada célula  $nc \in$  vizinhança  $V_\varepsilon(c)$  faça
7           tempPoint  $\leftarrow$  core-point  $\in nc$  mais próximo de  $p$ 
8           se tempPoint é mais próximo de  $p$  do que nearestPoint então
9             nearestPoint  $\leftarrow$  tempPoint
10          fim
11        fim
12        se nearestPoint = NULL então
13          | Rotular  $p$  como noise
14        senão
15          | Atribuir  $p$  ao agrupamento de nearestPoint
16        fim
17      fim
18    fim
19  fim
20 fim

```

4.3.4 Seleção de Representativos

As técnicas inseridas no método de classificação de características que foram apresentadas nas Seções 4.3.1 e 4.3.2 são aparentemente suficientes para superar as motivações iniciais deste trabalho. No entanto, por mais que o resultado seja um gráfico de dispersão, em que são delimitados agrupamentos – aqui representando características volumétricas – a manipulação de uma grande nuvem de pontos projetados como pontos nem sempre é um processo inteligível. Esta última etapa do método tem o propósito de simplificar o espaço exploratório e refinar as características volumétricas identificadas.

Técnicas de seleção de representativos são largamente utilizadas em visualização e em outras áreas da ciência para reduzir um conjunto $\mathbb{C}$ a um subconjunto $\mathbb{S}$ que melhor represente as instâncias originais. Neste trabalho, o processo de seleção de representativos atua tanto na diminuição da densidade visual do gráfico de dispersão quanto no refinamento e construção de uma FT especificada. A técnica *Sparse Spatial Selection* (PEDREIRA;

BRISABOA, 2007) foi escolhida para ser responsável por ambas as tarefas. *Sparse Spatial Selection* (SSS) é uma técnica de seleção de representativos proposta por Pedreira e Brisaboa (2007) capaz de processar grandes *datasets* em um baixo tempo de execução.

A técnica completa é apresentada no Algoritmo 8. O SSS localiza os pontos mais afastados entre si, tipicamente chamados de “representativos”, de um conjunto $\mathbb{C}$, relativamente a um fator de distância β que varia no intervalo $[0, 1]$. A execução do algoritmo acompanha o seguinte fluxo de instruções. O primeiro elemento $c_1 \in \mathbb{C}$ é adicionado ao subconjunto $\mathbb{S}$. Em seguida, o conjunto original é percorrido à procura de outros representativos. Tendo M como a distância máxima entre dois pontos s_a e $s_b \in \mathbb{S}$, um elemento c só é adicionado ao subconjunto $\mathbb{S}$, se $\text{dist}(c, s_i) \geq M\beta$.

Algoritmo 8: Seleção de representativos.

Entrada: $\mathbb{C}$ Saída: $\mathbb{S}$	/* Conjunto original */ /* Subconjunto de $\mathbb{S}$ */
--	--


```

1 início
2   M ← 0
3   S ← {p1} para cada pi ∈ P faça
4     se ∀s ∈ S, dist(pi, s) ≥ Mβ então
5       S ← S ∪ {pi}
6       se dist(pi, s) ≥ M então
7         M ← dist(pi, s)
8       fim
9     fim
10  fim
11 fim
  
```

Para regular o tamanho do subconjunto $\mathbb{S}$, o usuário deve alterar o fator de distância β , que quanto mais próximo de 0, favorece a escolha de um maior número de elementos e, em contrapartida, resulta em uma diminuição de representativos quando mais próximo de 1.

4.3.5 Classificação de Subcaracterísticas

Partindo da premissa que após a etapa de seleção de representativos, há um par que associa um conjunto de representativos $\mathbb{S}$ a um conjunto de pontos $\mathbb{C}$, desenvolvemos o Algoritmo 9 para encontrar subcaracterísticas, que relaciona cada ponto $p \in \mathbb{C}$ a um representativo $s \in \mathbb{S}$.

Após a execução do SSS e do Algoritmo 9, um conjunto de *voxels* está associado a cada representativo e, por sua vez, cada representativo está associado a um agrupamento.

Algoritmo 9: Identificação de subcaracterísticas em um agrupamento.

```

Entrada: pares  $\{ \mathbb{S}, \mathbb{C} \mid \mathbb{S} \subset \mathbb{C} \}$ 

/*  $\mathbb{C}$  pontos de um mesmo agrupamento */
/*  $\mathbb{S}$  subconjunto de representativos do conjunto  $\mathbb{C}$  */
/*  $n$  quantidade total de agrupamentos */

1 início
2   para cada par  $\{ \mathbb{S}, \mathbb{C} \mid \mathbb{S} \subset \mathbb{C} \}$  faça
3     para cada ponto  $p \in$  conjunto  $\mathbb{C}_i$  faça
4       | Associar  $p$  ao representativo  $\{ s \mid s \in \mathbb{S}, s \leftarrow \min(\text{dist}(\mathbb{S}, p)) \}$ 
5     fim
6   fim
7 fim

```

4.4 Interface de Navegação

A abordagem de exploração volumétrica proposta engloba a interação em uma interface formada por pontos dispostos em um gráfico de dispersão simplificado. A fim de evitar qualquer ambiguidade de interpretação, é importante frisar que cada ponto retrata um representativo e, como efeito, um conjunto de *voxels* é indiretamente associado a cada ponto. Alguns desses conjuntos estão dentro do mesmo agrupamento e, desta forma, vinculados a uma mesma característica.

4.4.1 Especificação Inicial da Função de Transferência e *Lookup Table*

De antemão, não há cor, opacidade ou qualquer outra distinção visual entre pontos de diferentes agrupamentos. Quando o método de classificação se encerra, uma especificação inicial da FT é gerada a partir dos resultados obtidos. Uma escala de cores estabelece uma distribuição de cores pelo gráfico de dispersão, de forma que todos os pontos (representativos) de um mesmo agrupamento sejam atribuídos a uma mesma cor. Uma ilustração da distribuição de cores por meio da escala do arco-íris é apresentada na Figura 30. A opacidade é também incorporada à estrutura, de modo que a definição seja composta tanto por uma informação de cor quanto uma informação de opacidade.

A estratégia utilizada para representação da FT especificada é apresentada na Figura 31. A abordagem utiliza duas estruturas: uma meta-LUT e uma LUT. A meta-LUT indexa todos os *voxels* do *dataset*, armazenando em cada linha um índice para o representativo ao qual o *voxel* é associado. A LUT indexa pontos e armazena em cada linha uma cor e uma opacidade. As estruturas de ambas as tabelas permanecem inalteradas durante a interação do usuário.

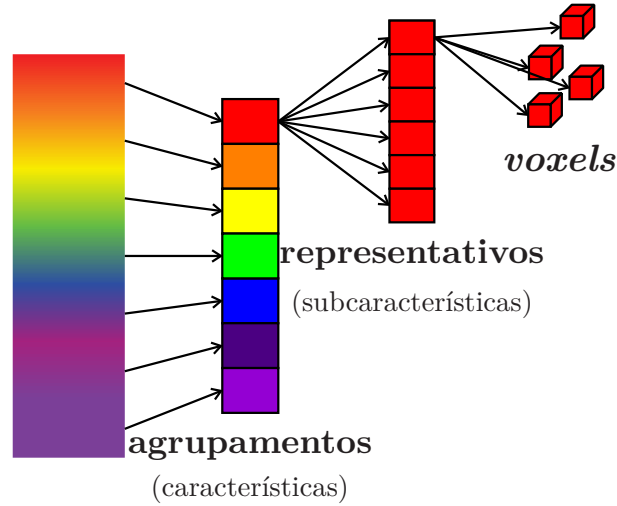
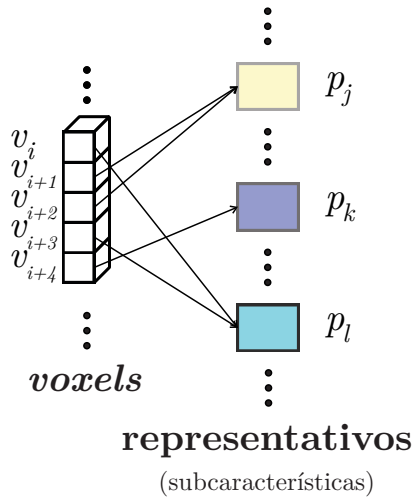


Figura 30 – Esquema de especificação inicial da função de transferência.

Figura 31 – *Lookup Table* para armazenamento da função de transferência especificada.

4.4.2 Navegação e Interação com o Volume

A ferramenta implementada para apoio da abordagem proposta neste capítulo é composta por duas janelas de interação. Uma primeira janela, na qual o gráfico de dispersão é exibido e uma outra, em que a imagem resultante da FT especificada é renderizada. O conceito *What You See Is What You Get* é adotado em cada ponto. A propriedade visual atribuída a um ponto é exatamente a mesma atribuída aos *voxels* associados. Quando o usuário seleciona um ponto qualquer, a região assinalada é enfatizada na visualização renderizada. Um exemplo da interação é apresentado na Figura 32. A Subfigura 32a é o gráfico de dispersão gerado. Uma seleção inicial (veja os pontos pontilhados na Subfigura 32b) revela o tecido ósseo e uma outra (veja os pontos pontilhados na Subfigura 32c), o tecido muscular de um pé.

O usuário pode ajustar¹ a especificação inicial, definindo cor e opacidade de qualquer

¹ Exemplos de edição. Disponível em: <<https://www.youtube.com/watch?v=bkZoAkxpN9I>>. Acesso

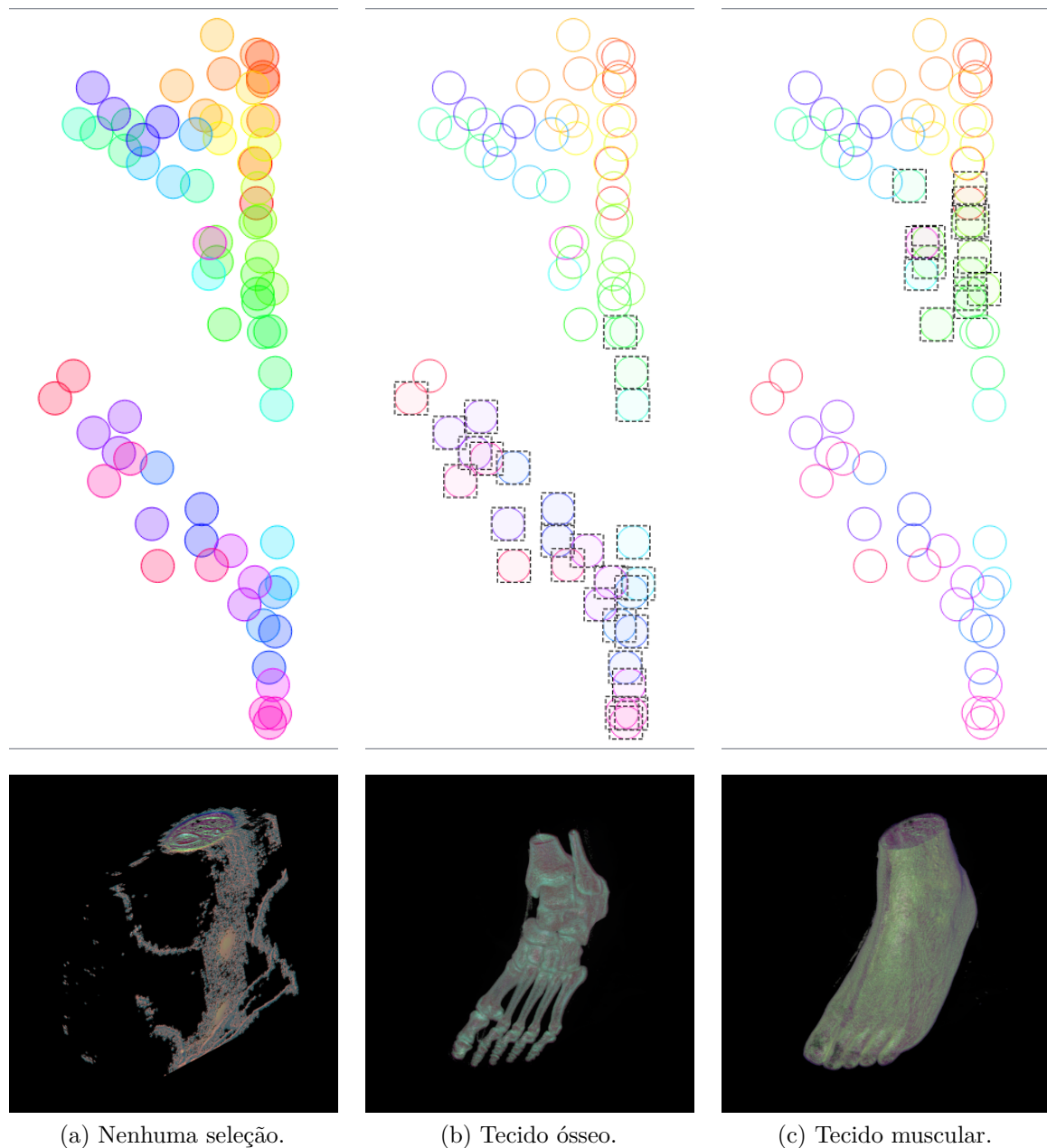


Figura 32 – Gráfico de dispersão de características e exploração do volume de um pé. Linha superior: seleções no gráfico de dispersão de características. Linha inferior: visualização decorrente das seleções.

ponto do gráfico. Ademais, também é possível modificar a posição do observador.

A exploração² do volume ocorre por meio da seleção de pontos pelo gráfico. Durante uma seleção, a opacidade dos pontos selecionados, simultaneamente das regiões correspondentes no volume, automaticamente aumenta, ao passo que a opacidade dos demais diminui. Deste modo, é importante destacar que a variação de opacidade é responsável

em: 12 de fevereiro de 2018.

² Exemplos de exploração. Disponível em: <<https://www.youtube.com/watch?v=RxkdDfs2F58>>. Acesso em: 12 de fevereiro de 2018.

por evidenciar detalhes de interesse, ao passo que suprimi demais regiões do volume.

Em relação a interação, há três tipos de elementos selecionáveis: pontos, agrupamentos e combinações manualmente predefinidas. Assim como na maioria dos aplicativos *desktop*, a ferramenta possibilita que tais elementos sejam selecionados pontualmente ou por área retangular. Em síntese, a exploração do volume pode ser conduzida por meio da ação de pressionar o mouse e arrastar gradativamente a área de seleção sobre os pontos do gráfico de dispersão. De forma imediata a essa ação, estruturas ocultas vão sendo reveladas.

4.5 Considerações Finais

A abordagem de exploração volumétrica proposta tem o propósito de propiciar uma manipulação efetiva de dados multidimensionais em Visualização Direta de Volumes. O objetivo diz respeito a concepção de um sistema que possa ser aplicado na prática. Para isso, a abordagem enfrenta simultaneamente os principais desafios que acometem a utilização desse tipo de ferramenta: a seleção de atributos volumétricos, o custo computacional, a classificação de características e a manipulação de espaços de alta dimensionalidade.

No Capítulo 5, são descritos experimentos realizados para validação da abordagem desenvolvida em relação aos objetivos citados e ao seu funcionamento em totalidade.

5 Resultados

Neste capítulo, descrevemos os experimentos realizados para avaliação da abordagem desenvolvida, particularmente, para o método de classificação de características volumétricas. Toda a discussão tem como tônica: a qualidade das características detectadas, o custo computacional do método de classificação e no modo como o esquema definitivamente assiste o usuário durante a exploração do volume. O conceito por trás de qualidade se refere tanto à separabilidade quanto à relevância das regiões identificadas.

5.1 Metodologia

A Tabela 2 discrimina os quatro *datasets* que são utilizados nos experimentos. Os *datasets* da cabeça, carpa e pé são providos pelo repositório da Universidade de *Erlangen-Nürnberg*¹. O *dataset* do pé está disponível no repositório da *U.S. National Library of Medicine*².

Tabela 2 – Especificação dos *datasets*.

<i>Dataset</i>	Tamanho	Total de <i>voxels</i>	Razão de aspecto dos <i>voxels</i>
Cabeça	$128 \times 256 \times 256$	8.388.608	$1,57774 \times 0,995861 \times 1,00797$
Carpa	$256 \times 256 \times 512$	33.554.432	$0,7812 \times 0,390625 \times 1$
Dente	$256 \times 256 \times 161$	10.551.296	$1 \times 1 \times 1$
Pé	$102 \times 247 \times 200$	5.038.800	$0,95 \times 0,95 \times 1$

Os dados volumétricos são constituídos apenas por valores escalares de intensidade, organizados em grades volumétricas regulares. A fim de conceber um espaço multidimensional, implementamos extratores de características. Além dos bem conhecidos gradiente de primeira e segunda ordem, dispomos de oito operadores de textura computados por descritores estatísticos locais: assimetria, desvio absoluto, desvio padrão, energia (homogeneidade), entropia, média, suavidade relativa e variância. Considerando que $f(x)$ é a função que define a distribuição de intensidade, v é a razão de aspecto dos *voxels* e $H(i)$ é um histograma local de tons de intensidade, os operadores são calculados conforme as fórmulas descritas na Tabela 3. A magnitude do gradiente é aproximada por diferenças centradas obtidas por expansão da série de Taylor. Caso os *voxels* sejam posicionados

¹ *University of Erlangen-Nürnberg's Volume Dataset Repository*. Disponível em: <<http://www9.informatik.uni-erlangen.de/External/vollib/>>. Acesso em: 07 de fevereiro de 2018.

² *U.S. National Library of Medicine*. Disponível em: <https://www.nlm.nih.gov/research/visible/fresh_ct.html>. Acesso em: 07 de fevereiro de 2018.

em alguma extremidade da grade – i.e., em casos de fronteira – diferenças regressivas ou progressivas são empregadas em substituição às centradas. O laplaciano aproxima o gradiente de segunda ordem. Nesse operador é adotado um procedimento semelhante ao da magnitude de gradiente para tratar fronteiras. Os operadores estatísticos são calculados a partir de histogramas de intensidade $H(i)$ localmente computados em regiões cúbicas ao redor de um *voxel* central. O histograma $H(i)$ representa uma região de tamanho $5 \times 5 \times 5$ que é deslocada unitariamente pela grade. Ambos os parâmetros relacionados à configuração da vizinhança, a dimensão e o passo de deslocamento, podem interferir no resultado do processo de classificação (CABAN; RHEINGANS, 2008). Optou-se pela configuração de região e deslocamento empregada por Caban e Rheingans (2008) que, segundo os próprios autores, se mostrou mais adequada para a maioria dos *datasets* testados no referido trabalho.

Tabela 3 – Operadores de extração de características.

Operador	Fórmula
Média	$\sum_{i=1}^n iH(i)$
Desvio Padrão	$\sqrt{\sum_{i=1}^n (\text{Média} - i)^2 H(i)}$
Variância	$\sum_{i=1}^n (\text{Média} - i)^2 H(i)$
Assimetria	$\frac{\sum_{i=1}^n (\text{Média} - i)^3 H(i)}{\text{Desvio Padrão}}$
Desvio Absoluto Médio	$\sum_{i=1}^n \text{Média} - i H(i)$
Energia	$\sum_{i=1}^n H^2(i)$
Entropia	$-\sum_{i=1}^n H(i) \log_2(H(i))$
Gradiente de Segunda Ordem	$\sum_{d=1}^3 \left(\frac{f(x_d + v_d) - 2 + f(x_d)f(x_d - v_d)}{v_d^2} \right)$
Magnitude do Gradiente	$\sqrt{\sum_{d=1}^3 \left(\frac{f(x_d + v_d) - f(x_d - v_d)}{2v_d} \right)^2}$
Suavidade Relativa	$1 - \frac{1}{1 - \text{Variância}}$

Os experimentos têm por propósito geral demonstrar como a abordagem de exploração se comporta em situações práticas de aplicação. Nesse cenário, avaliamos o número de parâmetros que são requeridos e principalmente a relevância das características descobertas.

A forma na qual a avaliação é conduzida é uma ameaça já assumida nos experimentos. Impreterivelmente, parte dos resultados obtidos são provenientes de configurações empíricas, guiadas por análises qualitativas. A especificação de FTs (PFISTER et al., 2001), o problema de automação e o campo de pesquisa de visualização (TORY; MOLLER, 2004) são condicionados à percepção humana e ao contexto em que tanto usuário quanto os dados manipulados estão inseridos. Diante dos fatores expostos, buscou-se utilizar *datasets* já consagrados e recorrentemente explorados por abordagens similares ao que é proposto para um melhor embasamento da discussão. A título de exemplo, alguns trabalhos dos seguintes autores que se enquadram nessa ideia: Pinto e Freitas (2007), Wang, Zhao e Kaufman (2012a), Ponciano, Seefeldt e Marroquim (2016) e Cai et al. (2017).

5.1.1 Aspectos de Implementação

A abordagem desenvolvida e outros dois componentes complementares foram combinados em uma ferramenta funcional de exploração volumétrica. O primeiro desses componentes é a técnica *Ray Casting*. A versão convencional do algoritmo foi implementada em CUDA para execução paralela em GPUs NVIDIA. A extração de atributos é o segundo componente. Assim como o processo de renderização, o procedimento também se beneficiou do poder de processamento de GPUs. Operadores de extração podem ser facilmente paralelizáveis, já que o cálculo de uma região local independe de regiões adjacentes. A única instrução que não é naturalmente paralelizável é o somatório. No caso dessa instrução, adotou-se uma estratégia de redução paralela (HARRIS, 2007) baseada no paradigma de divisão e conquista.

As técnicas incluídas na abordagem – *FastMap*, DBSCAN 2D e SSS – foram implementadas sequencialmente em CPU por meio da linguagem C++. O *framework* OpenCV foi empregado para carregamento do *dataset*. Além disso, os *frameworks* OpenGL e Qt foram utilizados para construção da ferramenta e da interface de exibição.

É possível acelerar o DBSCAN com matrizes de distâncias, mas nenhuma estrutura desse tipo foi adotada. Devido ao tamanho dos *datasets*, o consumo de memória exigido torna a estratégia não viável, já que pode exceder o total de memória principal disponível. O mesmo problema prejudica as versões paralelas dessa técnica.

5.1.2 Design dos Experimentos

O passo de caminamento do *Ray Casting* foi definido em 0,25 unidade de medida. Modelos de interpolação trilinear e iluminação Blinn-Phong (BLINN, 1977) também foram incluídos para evidenciar o aspecto de profundidade e propiciar um elevado nível de detalhes.

Dados volumétricos multidimensionais foram obtidos mediante a execução de

extratores de características. Antes da aplicação do método de classificação, os valores dos atributos foram normalizados.

As medidas de tempo apresentadas foram aferidas por meio de uma média de cinco execuções. O computador utilizado nos experimentos tem a seguinte configuração:

- CPU Intel Core i5-7200U 2.5Ghz;
- 8 GB DDR4 RAM;
- NVIDIA GeForce 740MX 2GB GDDR5 RAM com 512 CUDA *cores*;
- Windows 10 64 bits.

No texto, assumimos a seguinte simbologia para os parâmetros: α para o fator de decisão entre entropia e correlação da heurística de seleção de atributos volumétricos; β para o fator de distância do *Sparse Spatial Selection*; ε para o raio de consulta do DBSCAN; e *minPts* para o número mínimo de pontos desse mesmo algoritmo.

5.2 Experimentos

A heurística de seleção de atributos foi inicialmente executada em cada um dos *datasets*. Considerando os atributos descritos na Tabela 2 e $\alpha = 0,5$, os resultados produzidos são apresentados na Tabela 4. A intensidade foi separada no *ranking*, visto que é o dado fonte a partir do qual os demais atributos são derivados.

A etapa que envolve a execução da heurística é preliminar e opcional. Ainda que forneça informações para indicar uma possível composição da assinatura dos *voxels*, ainda é de total responsabilidade do usuário definir a combinação de atributos. Como já mencionado na Seção 4.2, o *ranking* é na verdade apenas uma orientação ao usuário para construção do domínio da função de transferência.

5.2.1 Espaços Multidimensionais

No primeiro experimento, aplicamos o método de classificação nos *datasets* já com atributos volumétricos adicionais. A definição dos parâmetros β , ε e *minPts* ocorreu de forma empírica. No que tange o primeiro parâmetro, o ajuste é inteligível. Há uma relação inversa entre o valor de β e a quantidade de vértices por agrupamento, tal como apontado por Pedreira e Brisaboa (2007). Por outro lado, a determinação dos parâmetros do DBSCAN é uma tarefa trabalhosa (SCHUBERT et al., 2017). Seguindo algumas diretrizes (ESTER et al., 1996; SCHUBERT et al., 2017), limitamos *minPts* a um número pequeno e variamos ε até encontrar resultados que permitissem uma separação cuidadosa das estruturas, mas que não fossem acometidos por uma grande fragmentação. Em outras

Tabela 4 – *Ranking* de atributos candidatos à composição da assinatura dos *voxels* para $\alpha = 0,5$.

	Cabeça	Carpa	Dente	Pé
#0	Intensidade	Intensidade	Intensidade	Intensidade
#1	Assimetria	Assimetria	Assimetria	Assimetria
#2	Desvio Absoluto	Entropia	Desvio Absoluto	Entropia
#3	Gradiente de 2a Ordem	Gradiente de 2a Ordem	Variância	Gradiente de 2a Ordem
#4	Média	Variância	Entropia	Variância
#5	Suavidade Relativa	Média	Gradiente de 2a Ordem	Desvio
#6	Variância	Desvio Absoluto	Desvio Padrão	Desvio Padrão
#7	Entropia	Desvio Padrão	Suavidade Relativa	Suavidade Relativa
#8	Desvio Padrão	Suavidade Relativa	Média	Média
#9	Energia	Energia	Energia	Energia
#10	Magnitude de Gradiente	Magnitude de Gradiente	Magnitude de Gradiente	Magnitude de Gradiente

palavras, buscamos uma configuração que isolasse as características com um menor número de agrupamentos.

As imagens resultantes do experimento não são associadas a um único agrupamento, mas a uma seleção manual sobre o gráfico. Agrupamentos próximos podem teoricamente pertencer a uma característica maior. Ademais, o método também foi utilizado para reconhecer *voxels* de fundo e, assim, nenhum pré-processamento da grade teve de ser realizado.

A Figura 33 apresenta resultados da aplicação do método de classificação no *dataset* da cabeça. A partir de uma exploração do gráfico de dispersão, são reveladas cinco grandes estruturas: dentes, pele, seio paranasal, tecido muscular e tecido ósseo. Ao todo, foram produzidos 39 agrupamentos. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,2$ e $minPts = 4$. Os seguintes atributos foram selecionados: intensidade, magnitude de gradiente e assimetria. Em discrepância ao resultado do *ranking*, uma maior separação das estruturas só foi alcançada com a adição da magnitude de gradiente. Segundo a heurística, a magnitude de gradiente é o atributo menos indicado para ser adicionado ao espaço da

FT. Desta forma, ao seguir o que foi apontado – ou seja, a adição de todos atributos até a magnitude de gradiente – houve uma elevada concentração de ruído, comprometendo o resultado.

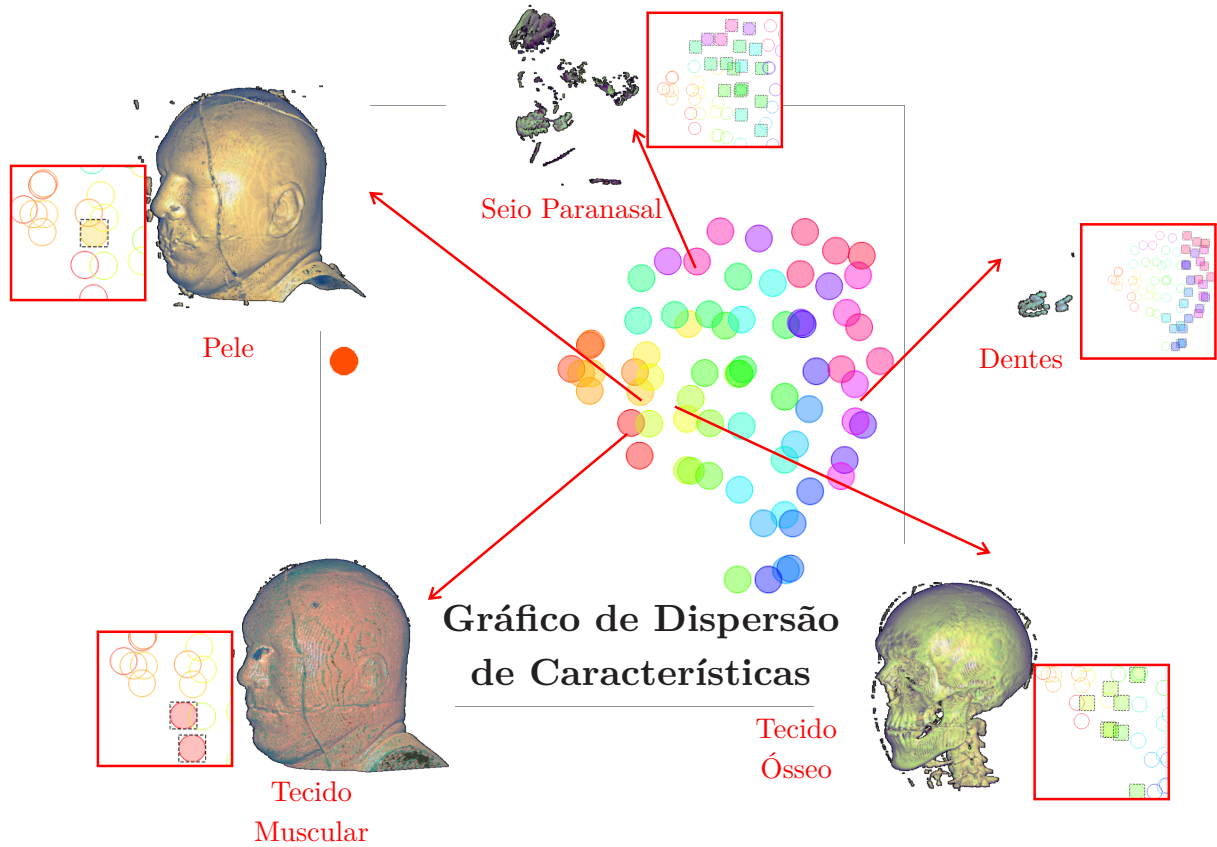


Figura 33 – *Dataset* da cabeça. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,2$ e $minPts = 4$. Atributos: intensidade, assimetria e magnitude de gradiente.

O *dataset* da carpa é apresentado na Figura 34. As principais características classificadas são: bexiga natatória e escamas, esqueleto e tecido muscular. Configuração de parâmetros: $\beta = 0,75$, $\varepsilon = 0,1$ e $minPts = 5$. No gráfico de dispersão, há 15 agrupamentos. Apenas a combinação da intensidade com o primeiro atributo indicado na Tabela 4, a assimetria, já proporcionou uma classificação mais sofisticada, mesmo que não tenha sido possível isolar a bexiga natatória das escamas.

Estruturas presentes no *dataset* do dente são apresentadas na Figura 35. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,175$ e $minPts = 5$. O DBSCAN teve como resultado 36 agrupamentos, que podem ser organizados em quatro características principais: coroa, dentina, esmalte e polpa. Novamente, a heurística de seleção de atributos falhou. As características só foram efetivamente destacadas com uma combinação de parâmetros diferente da apontada pelo *ranking*. Atributos volumétricos selecionados: intensidade, assimetria, magnitude de gradiente e suavidade relativa.

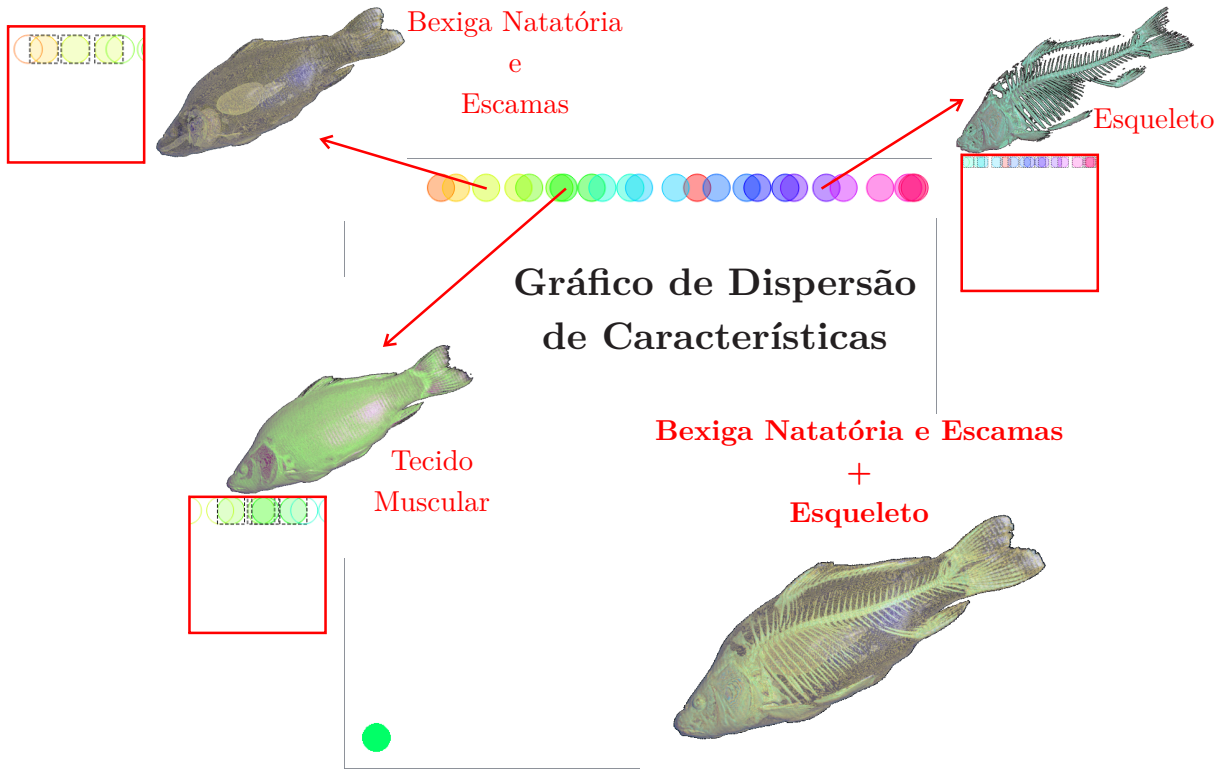


Figura 34 – *Dataset* da carpa. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $\minPts = 5$. Atributos: intensidade e assimetria.

O gráfico de dispersão de características resultante da aplicação do método no *dataset* do pé é apresentado na Figura 36. Estruturas em evidência: pele, tecido muscular e tecido ósseo. Os 29 agrupamentos detectados são provenientes de uma combinação da intensidade com os dois primeiros atributos do *ranking*: assimetria e entropia Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,15$ e $\minPts = 4$.

Nenhum experimento com intuito de aferir como a dimensionalidade afeta a classificação de características foi conduzido. Como já discutimos, a assinatura dos *voxels* é, em si, o fator responsável por isso.

5.2.2 Espaços Unidimensionais

Em um segundo experimento, investigamos como o método de classificação opera sobre dados unidimensionais de intensidade. Para analisar tal comportamento, os parâmetros $\beta = 0,75$; $\varepsilon = 0,1$ e $\minPts = 4$ tiveram os valores fixados. Os *datasets* foram, em unanimidade, divididos em 14 agrupamentos. A Figura 37 apresenta as regiões correspondentes a cada um desses agrupamentos no *dataset* da cabeça. As estruturas destacadas do *dataset* da carpa são apresentadas na Figura 38 e do *dataset* do dente, na Figura 39.

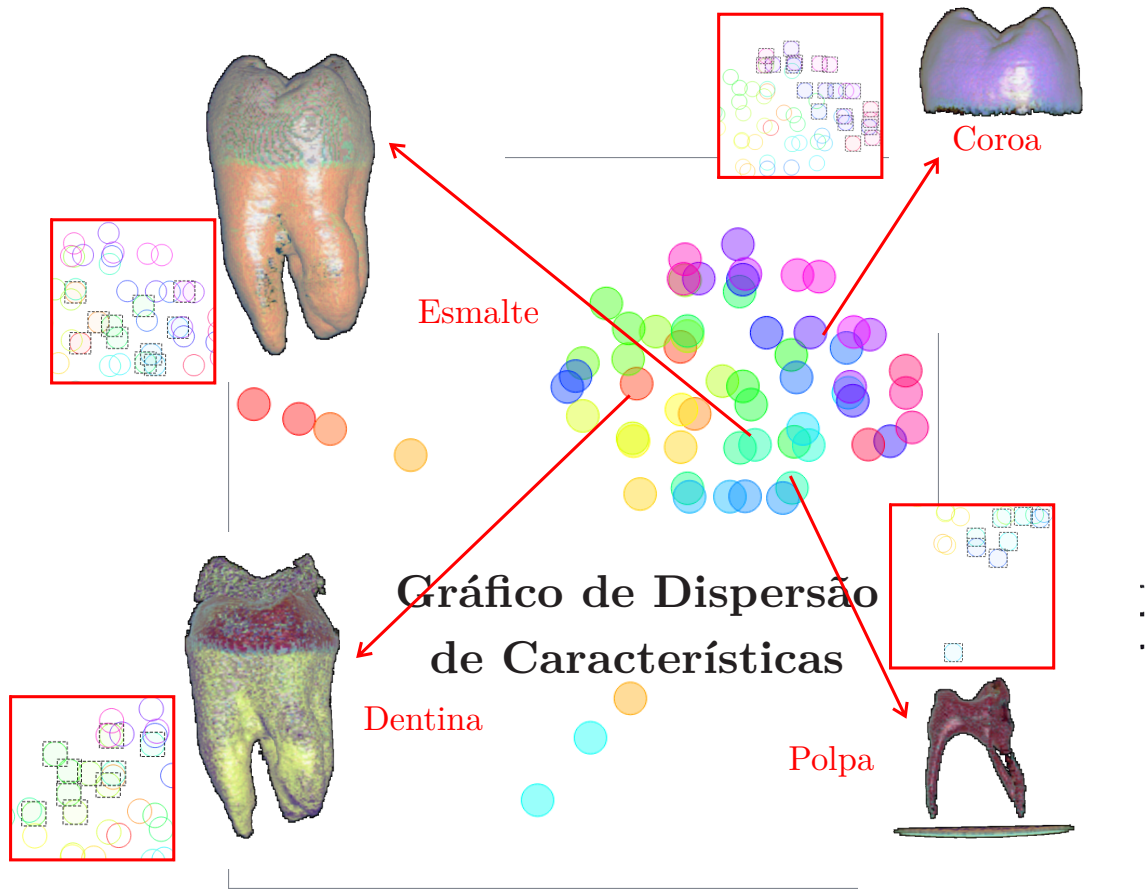


Figura 35 – *Dataset* do dente. Configuração de parâmetros: $\beta = 0,75$, $\varepsilon = 0,175$ e $minPts = 5$. Atributos: intensidade, assimetria, magnitude de gradiente e suavidade relativa.

Por fim, características reveladas no *dataset* do pé são apresentadas na Figura 40. À parte da separabilidade de características, os resultados demonstram que é possível estender o método para lidar com domínios unidimensionais.

É visualmente possível constatar que atuando apenas sobre a intensidade algumas estruturas relevantes não são meticulosamente isoladas. Algumas dessas estruturas são o seio paranasal no *dataset* da cabeça e a polpa no *dataset* do dente. Ainda assim, os resultados obtidos demonstraram que o método de classificação pode ser estendido para manipular dados unidimensionais.

5.2.3 Tempo de execução

Os tempos de execução do método de classificação decorrente do experimento descrito na Subseção 5.2.1 são apresentados na Tabela 5. O tempo médio de execução do método foi de 27,97 segundos para os *datasets* testados com as configurações da Subseção 5.2.1. Como pode ser observado, o maior *dataset* analisado consumiu mais tempo de execução.

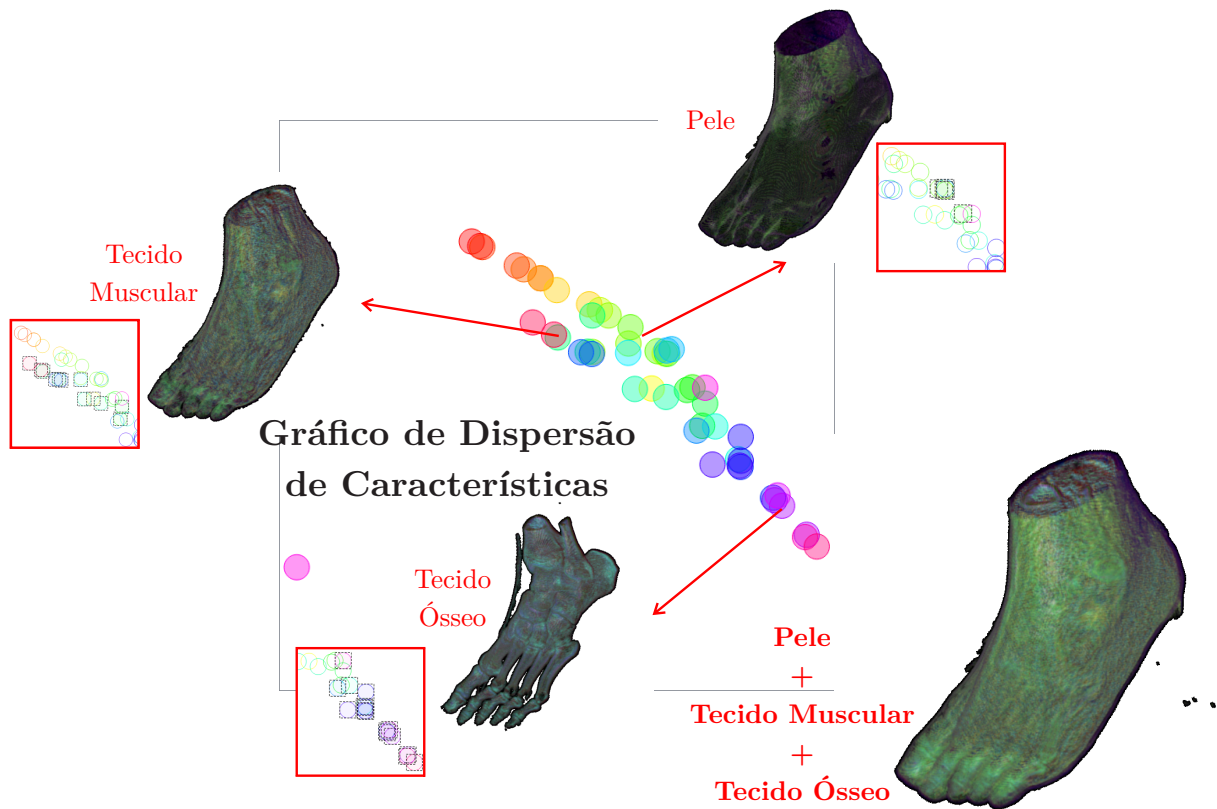


Figura 36 – *Dataset* do pé. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,15$ e $minPts = 4$. Atributos: intensidade, assimetria e entropia.

Tabela 5 – Tempo de execução do método de classificação de características para os *datasets* da cabeça, carpa, dente e pé com os atributos dos experimentos da Subseção 5.2.1.

<i>Dataset</i>	Atributos	Tempo de execução (em segundos)			
		<i>FastMap</i>	DBSCAN	SSS	Total
Cabeça	3	3,66	4,37	1,86	9,89
Carpa	2	0	35,65	10,96	46,62
Dente	4	1,61	4,7	1,32	7,64
Pé	3	2,24	7,82	0,65	10,71

Em consonância à complexidade de tempo dos algoritmos, foi possível constatar que o DBSCAN é, na prática, a etapa mais custosa do método. Todavia, resalta-se que em situações específicas, nas quais o número de dimensões é suficientemente grande ou que ocorre um grande número, o *FastMap* bem como o SSS podem exigir um maior custo computacional do que o DBSCAN.

Um último experimento foi realizado para avaliar como aumento da dimensionalidade interfere no desempenho da abordagem. Nesse experimento, adicionamos todos os atributos volumétricos disponíveis na assinatura dos *voxels*. Os valores de β , ε e $minPts$ foram mantidos em relação às configurações descritas na Subseção 5.2.1. O novo tempo de

execução é apresentado na Tabela 6. Essa tabela apresenta ainda o número de agrupamento (características) produzido. O tempo de execução foi em geral pouco afetado, com exceção do *dataset* da carpa que fez a técnica consumir mais do que o dobro do tempo em comparação à Tabela 5. Tendo em vista que o espaço em que o DBSCAN opera é sempre bidimensional, a variação de tempo pode ser explanada pela variação dos dados, mesmo que o valor dos parâmetros tenha sido fixado, a relação entre esses valores e domínio mudou.

Tabela 6 – Tempo de execução do método de classificação de características para os *datasets* da cabeça, carpa, dente e pé com 11 atributos volumétricos.

<i>Dataset</i>	Agrupamentos	Tempo de execução (em segundos)			
		<i>FastMap</i>	DBSCAN	SSS	Total
Cabeça	69	3,94	3,59	1,19	8,72
Carpa	278	16,98	75,90	5,15	98,03
Dente	50	2,31	5,34	1,34	8,99
Pé	178	2,60	7,90	0,70	10,92

5.3 Discussão

Os experimentos realizados permitem afirmar que a abordagem desenvolvida é um esquema de exploração para a edição de FTs e manipulação de dados volumétricos multidimensionais. As estratégias empregadas simplificam o processo, automatizando parte do trabalho. Há algumas esferas de ação ainda incumbidas ao utilizador. Contudo, os parâmetros livres são na verdade mecanismos de controle para acolhimento do *feedback* do usuário. É válido lembrar que a percepção humana é particularmente primordial para o processo de exploração volumétrica (KÖNIG; GRÖLLER, 2001).

5.3.1 Seleção de Atributos

A heurística de seleção de atributos falhou para os *datasets* da cabeça e do dente. Características pertinentes deixaram de ser precisamente isoladas. Essa constatação se fundamenta em uma análise subjetiva baseada em outros trabalhos da literatura. Em ambos os *datasets*, a ordem de atributos que devem ser adicionados ao espaço não destaca de forma apurada algumas estruturas presentes nos volumes. Um atributo específico, a magnitude do gradiente, teve de ser incorporado para que a execução do método de classificação fosse bem sucedida.

É importante ter em mente que avaliação dos resultados é subjetiva. Em face da discordância entre a essência qualitativa da exploração e a quantitativa da heurística de seleção de atributos – veja que as pontuações são baseadas em valores estatísticos – nem

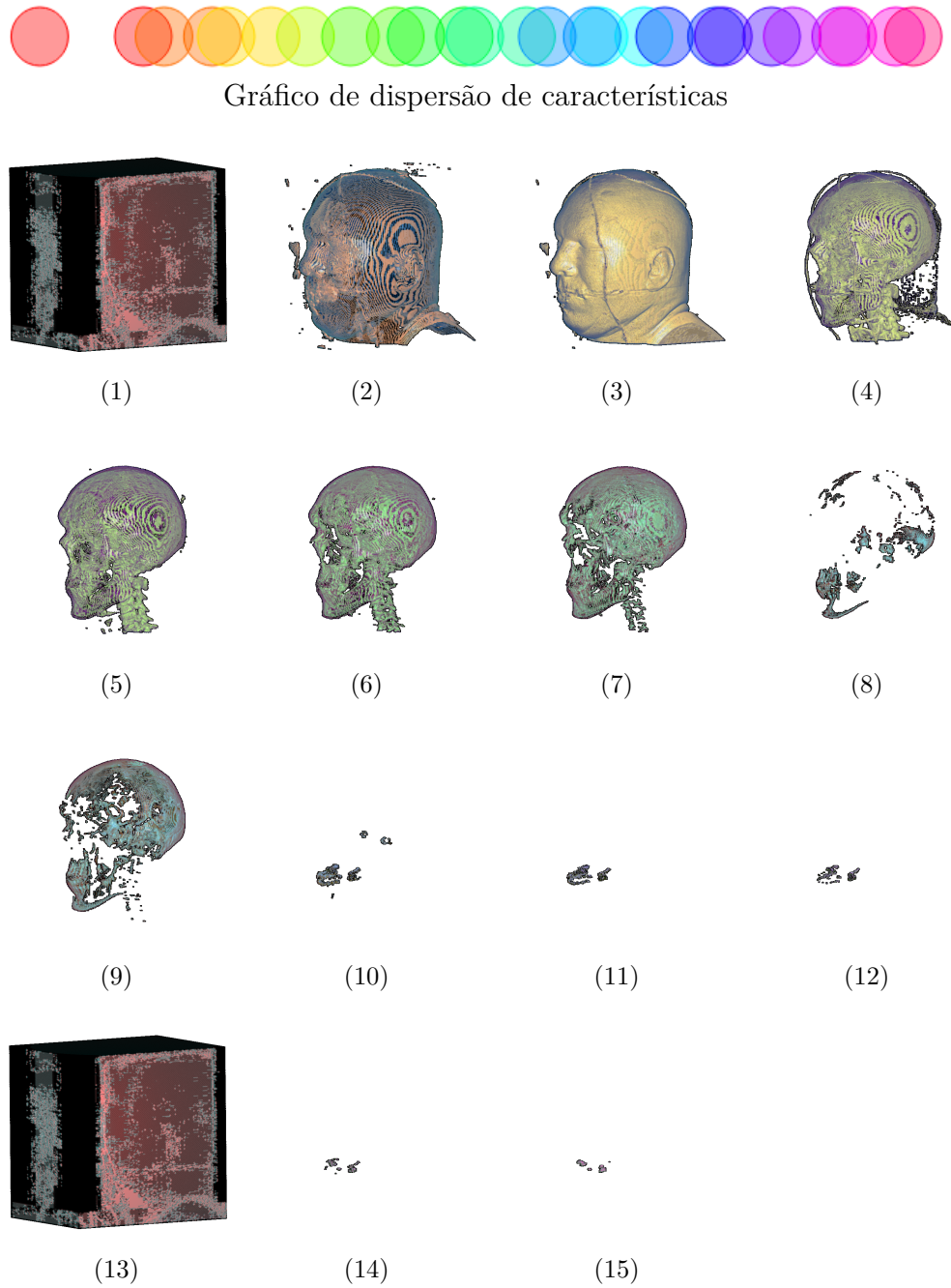


Figura 37 – *Dataset* da cabeça e 14 características classificadas a partir da intensidade. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $minPts = 4$.



Gráfico de dispersão de características.

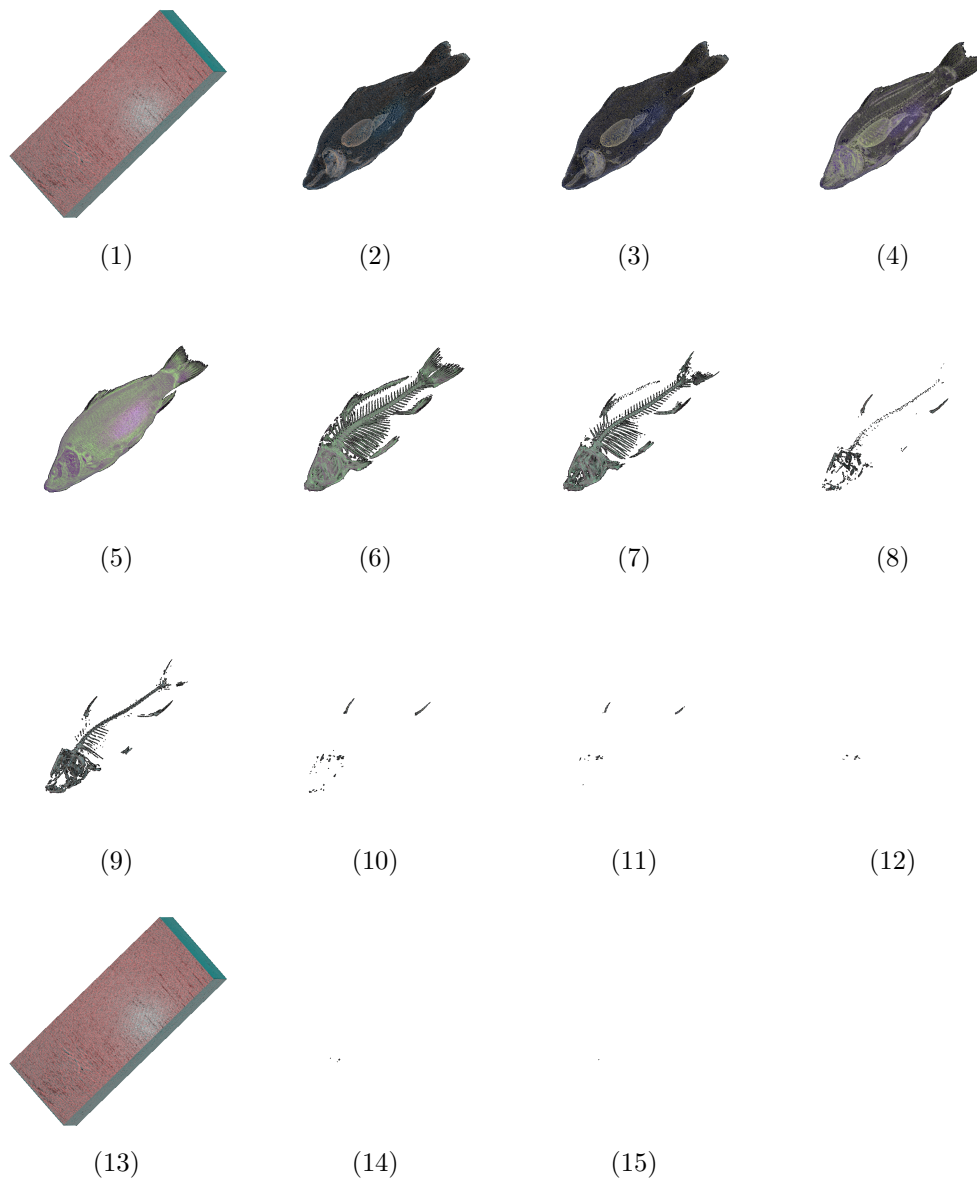


Figura 38 – *Dataset* da carpa e 14 características classificadas a partir da intensidade.
 Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $minPts = 4$.

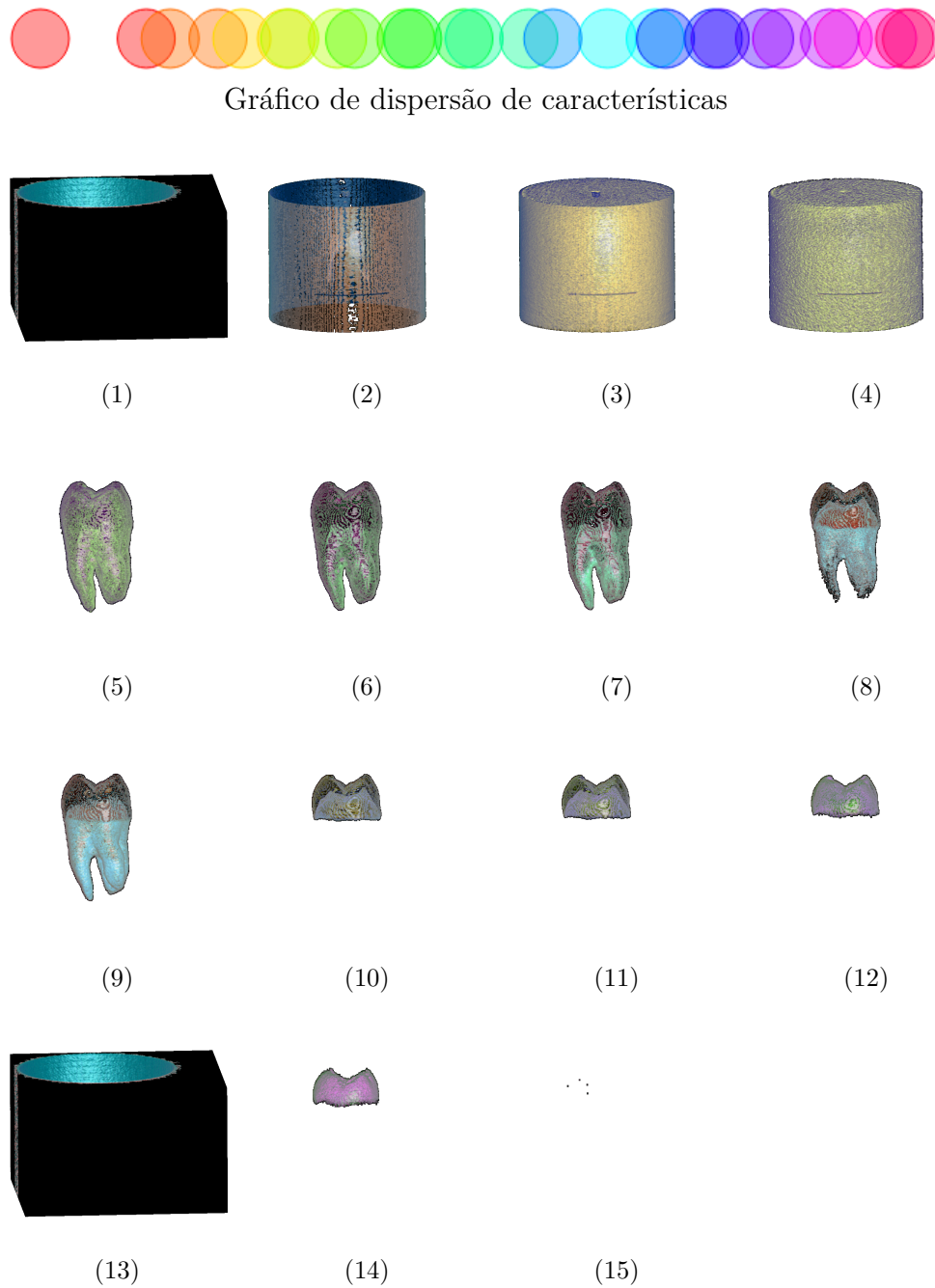


Figura 39 – *Dataset* do dente e 14 características classificadas a partir da intensidade.
Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $minPts = 4$.

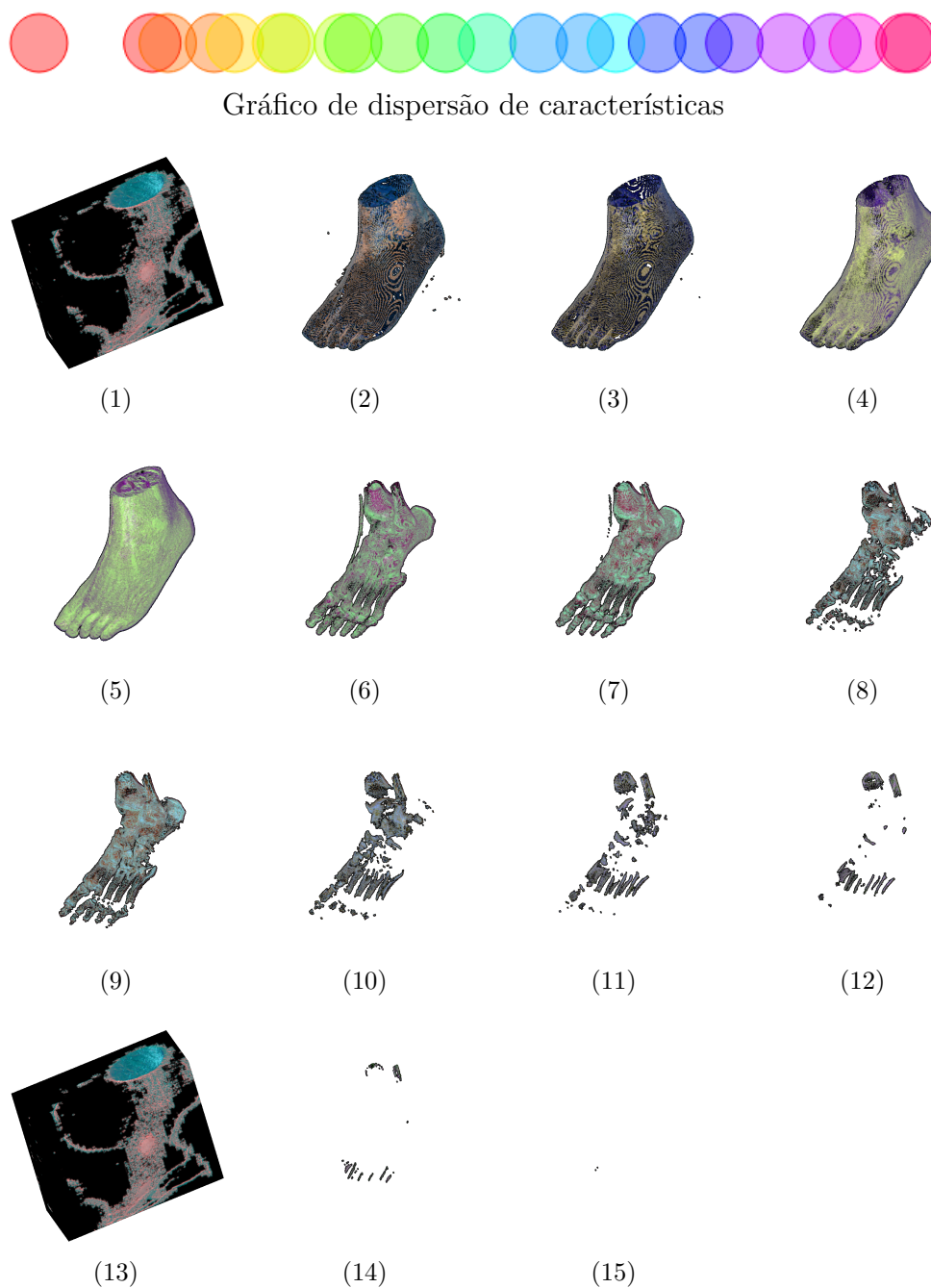


Figura 40 – *Dataset* do dente e 14 características classificadas a partir da intensidade. Configuração de parâmetros: $\beta = 0,75$; $\varepsilon = 0,1$ e $minPts = 4$.

sempre produzem classificações satisfatórias. Estruturas distintas podem não ser separadas, mesmo que isso não ocorra, pode ser gerada uma grande quantidade de ruído.

Apesar de tudo, é possível afirmar que a técnica é particularmente útil quando não há um conhecimento prévio por parte do usuário em relação ao volume explorado. Como já mencionado na Seção 4.2, métodos para seleção de atributos ainda são poucos explorados em FTs e mesmo não se limitando ao referido contexto, a própria definição de estratégias não supervisionadas para esse propósito é uma questão em aberto na literatura (QIAN; ZHAI, 2013).

5.3.2 Exploração do Volume e Configuração de Parâmetros

Para realmente validar a abordagem desenvolvida, é esperado que agrupamentos produzidos pelo método de classificação assegurem uma identificação acurada de objetos relevantes contidos no volume. Além disso, é esperado que não seja trabalhoso reconhecer no gráfico de dispersão de características cada um desses objetos.

A subordinação dos dois objetivos enunciados à seleção de atributos já foi discutida na Subseção 5.3.1. No entanto, os parâmetros dos algoritmos SSS (β) e DBSCAN (ε e $minPts$) também têm parte da responsabilidade.

O entendimento da forma em que o parâmetro β afeta a exploração do volume é transparente. Sem grandes pormenores, uma diminuição do fator de distância beneficia o aumento de vértices (subcaracterísticas) em todos os agrupamentos. Um resultado contrário é conquistado mediante o aumento do valor do parâmetro.

As variáveis de entrada $minPts$ e ε influenciam mais ativamente a abordagem. O número mínimo de pontos é atrelado à detecção de ruído (SCHUBERT et al., 2017). Comparado ao raio de consulta, a sensibilidade do processo de classificação ao parâmetro é menor. Durante os experimentos, assumiu-se uma referência inicial onde foi o estabelecido valores baixos. De acordo com Ester et al. (1996), $minPts = 4$ já uma configuração suficiente para a maioria dos *datasets*. O raio de consulta é mais difícil de ser definido (SCHUBERT et al., 2017). Tipicamente, a opção por raios de consulta menores tem como resultado um maior fragmentação do volume, ou seja, um maior número de agrupamentos e, consequentemente, de características classificadas. Proporcionalmente, o aumento do valor de ε proporciona o surgimento de agrupamentos com uma maior concentração de regiões e, em alguns casos, a junção de características volumétricas. Aqui, é importante frisar que a adoção de cada uma das técnicas propaga as peculiaridades inerentes à abordagem de exploração.

Fixar valores para cada um dos parâmetros é uma tarefa que cabe ao usuário e faz parte do processo de exploração volumétrica. O baixo custo computacional da abordagem torna possível retroceder e refinar os valores definidos. Além disso, constatamos que a configuração $minPts = 4$ e $\varepsilon = 0,1$ é um bom ponto de partida.

O argumento de que a execução do *FastMap* preliminarmente ao DBSCAN compromete severamente a acurácia do método de agrupamento não foi contrariado pelos experimentos realizados. Como já apontado por Fodor (2002), o método tem por vantagem preservar as distâncias entre as instâncias e concomitantemente assegurar um baixo tempo de execução.

No que diz respeito a navegação pelo gráfico de dispersão, regiões vizinhas representam características com uma maior semelhança entre si. Uma forma simples de analisar os objetos volumétricos reconhecidos consiste em fazer uma seleção gradativa pelo gráfico de dispersão.

5.4 Considerações Finais

Os experimentos apresentados permitiram a validação deste trabalho. A abordagem mostrou-se uma promissora abstração para a definição de FTs multidimensionais e um guia robusto para a manipulação de dados volumétricos de complexidade desconhecida.

6 Conclusão

Funções de transferência são um dos desafios mais antigos de Visualização Direta de Volumes e, apesar do esforço da comunidade científica, a crescente utilização de *datasets* maiores e complexos reaviva essas dificuldades. Esse cenário abrange o custo computacional, a busca por um processo de edição mais intuitivo e a promoção de uma classificação mais apurada de objetos volumétricos. Há diferentes abordagens que automatizam a manipulação de espaços multidimensionais, mas muitas dessas exigem um elevado consumo de recursos computacionais. Além disso, poucos trabalhos tratam a composição de assinatura dos *voxels* como um fator decisivo no poder de classificação de características.

6.1 Contribuições e Limitações

A principal contribuição deste trabalho é fornecer uma abordagem de exploração volumétrica flexível que é capaz de simplificar a manipulação de espaços multidimensionais e, ainda assim, ser aplicável a situações reais de investigação. Mesmo em uma execução puramente sequencial, a complexidade de tempo do método de classificação e da construção do espaço de edição tem por limite prático $O(n \log n)$. Isso é garantido pela utilização do *FastMap* para redução dimensional e pela versão do DBSCAN proposta por [Gunawan e Berg \(2013\)](#) para agrupamento. Em relação à utilização do SSS, o objetivo de resumir o espaço de edição sem comprometer o tempo de execução foi atingido.

Uma contribuição secundária deste trabalho é a heurística de seleção de atributos. Apesar de alguns testes apresentados indicarem resultados não satisfatórios, a adaptação da técnica de [Zhou, Schott e Hansen \(2012\)](#) é uma possível alternativa quando não há um sólido conhecimento prévio por parte do utilizador a respeito do *dataset* explorado. Além disso, é importante notar que a avaliação dos resultados ocorre sob uma ótica subjetiva.

Ainda que não seja intimamente relacionada ao escopo deste trabalho, uma das maiores limitações é a avaliação de como o ajuste de parâmetros do DBSCAN afeta diretamente a qualidade das características volumétricas classificadas. Cabe ressaltar que essa ação é uma questão em aberto recorrentemente discutida na literatura ([KARAMI; JOHANSSON, 2014](#)). Concomitantemente a isso, é possível avaliar como adotar algumas estratégias propostas nesse campo para guiar a definição dessas variáveis de entrada.

Uma limitação ainda associada ao método de classificação é a necessidade de se analisar a curva de aprendizagem e o comportamento de usuários reais em uma ferramenta de exploração que implemente a abordagem desenvolvida.

No que diz respeito à técnica de seleção de atributos volumétricos, há algumas

limitações que devem ser apontadas. O nível de auxílio do guia proposto ainda é insuficiente. Desta forma, é importante admitir que um elevado conhecimento prévio é requisitado por parte dos usuários. Adicionalmente, há também a ausência de uma análise mais detalhada de como a pontuação se vincula aos resultados produzidos.

6.2 Trabalhos Futuros

Os trabalhos futuros aqui propostos visam enfrentar as limitações anteriormente constatadas. Um possível trabalho tem como objetivo comparar discriminadamente a abordagem de exploração volumétrica desenvolvida com outras estratégias descritas nesse campo de atuação, bem como por em prática experimentos para investigar diretamente a ação dos usuários, tanto em relação à familiarização como em relação à percepção de detalhes volumétricos inseridos nos *datasets*.

Um outro experimento a ser realizado é a avaliação de classificadores com intuito de comprovar que, de fato, a redução dimensional não distorce os resultados obtidos pelo DBSCAN. Nesse sentido, é importante ter em mente que o custo computacional das versões da técnica para espaços de maior dimensionalidade são atualmente uma barreira a esse experimento.

Na técnica de seleção de características, uma primeira ideia a ser considerada é a substituição dos operadores de entropia e de correlação múltipla, respectivamente de [Shannon \(1948\)](#) e de [Pearson \(1901\)](#), por outras medidas mais sofisticadas.

Além de tudo isso, é importante investigar se outras técnicas propostas para seleção de atributos em conjuntos não rotulados podem ser utilizadas a fim de se obter *rankings* que assinalem configurações de assinatura de *voxels* satisfatórias, assumindo que o conceito se refere ao destaque de estruturas e materiais relevantes.

Referências

- ABDI, H. Multiple correlation coefficient. Citeseer, 2007. Citado na página 47.
- ARENS, S.; DOMIK, G. A survey of transfer functions suitable for volume rendering. In: *Proceedings of the 8th IEEE/EG International Conference on Volume Graphics*. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2010. (VG'10), p. 77–83. ISBN 978-3-905674-23-1. Disponível em: <<http://dx.doi.org/10.2312/VG/VG10/077-083>>. Citado 5 vezes nas páginas 17, 27, 28, 29 e 42.
- BAJAJ, C. L.; PASCUCCI, V.; SCHIKORE, D. R. The contour spectrum. In: IEEE COMPUTER SOCIETY PRESS. *Proceedings of the 8th conference on Visualization'97*. [S.l.], 1997. p. 167–ff. Citado na página 38.
- BEYER, J.; HADWIGER, M.; PFISTER, H. A survey of gpu-based large-scale volume visualization. IEEE Visualization and Graphics Technical Committee (IEEE VGTC), 2014. Citado 2 vezes nas páginas 25 e 43.
- BEYER, J.; HADWIGER, M.; PFISTER, H. State-of-the-art in gpu-based large-scale volume visualization. In: WILEY ONLINE LIBRARY. *Computer Graphics Forum*. [S.l.], 2015. v. 34, n. 8, p. 13–37. Citado na página 16.
- BLINN, J. F. Models of light reflection for computer synthesized pictures. In: ACM. *ACM SIGGRAPH Computer Graphics*. [S.l.], 1977. v. 11, n. 2, p. 192–198. Citado na página 66.
- BODDY, R.; SMITH, G. *Statistical methods in practice: for scientists and technologists*. [S.l.]: John Wiley & Sons, 2009. Citado na página 46.
- BRODLIE, K. W. et al. *Scientific visualization: techniques and applications*. [S.l.]: Springer Science & Business Media, 2012. Citado na página 20.
- CABAN, J. J.; RHEINGANS, P. Texture-based transfer functions for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, v. 14, n. 6, p. 1364–1371, Nov 2008. ISSN 1077-2626. Citado 3 vezes nas páginas 32, 40 e 65.
- CAI, L. et al. A two-level clustering approach for multidimensional transfer function specification in volume visualization. *The Visual Computer*, v. 33, n. 2, p. 163–177, Feb 2017. ISSN 1432-2315. Disponível em: <<https://doi.org/10.1007/s00371-015-1167-y>>. Citado 6 vezes nas páginas 17, 18, 40, 41, 43 e 66.
- CHENG, Y. Mean shift, mode seeking, and clustering. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 17, n. 8, p. 790–799, 1995. Citado na página 40.
- CORREA, C.; MA, K.-L. Size-based transfer functions: A new volume exploration technique. *IEEE transactions on visualization and computer graphics*, IEEE, v. 14, n. 6, p. 1380–1387, 2008. Citado 2 vezes nas páginas 30 e 31.
- CORREA, C. D.; MA, K.-L. Visibility-driven transfer functions. In: IEEE. *2009 IEEE Pacific Visualization Symposium*. [S.l.], 2009. p. 177–184. Citado 2 vezes nas páginas 38 e 39.

- DREBIN, R. A.; CARPENTER, L.; HANRAHAN, P. Volume rendering. In: *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY, USA: ACM, 1988. (SIGGRAPH '88), p. 65–74. ISBN 0-89791-275-6. Disponível em: <<http://doi.acm.org/10.1145/54852.378484>>. Citado na página 27.
- DY, J. G.; BRODLEY, C. E. Feature selection for unsupervised learning. *Journal of machine learning research*, v. 5, n. Aug, p. 845–889, 2004. Citado na página 45.
- ELER, D. M. *Rendering híbrido: mapeamento de volumes sobre superfícies*. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2006. Citado na página 22.
- ELVINS, T. T. A survey of algorithms for volume visualization. *SIGGRAPH Comput. Graph.*, ACM, New York, NY, USA, v. 26, n. 3, p. 194–201, ago. 1992. ISSN 0097-8930. Disponível em: <<http://doi.acm.org/10.1145/142413.142427>>. Citado 4 vezes nas páginas 16, 20, 21 e 22.
- ENCARNACAO, L. M. et al. Future directions in computer graphics and visualization: From CG&A's Editorial Board. *IEEE Computer Graphics and Applications*, v. 35, n. 1, p. 20–32, Jan 2015. ISSN 0272-1716. Citado na página 25.
- ESTER, M. et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Kdd*. [S.l.: s.n.], 1996. v. 96, n. 34, p. 226–231. Citado 4 vezes nas páginas 51, 53, 67 e 78.
- FALOUTSOS, C.; LIN, K.-I. *FastMap: A Fast Algorithm for Indexing, Data-mining and Visualization of Traditional and Multimedia Datasets*. New York, NY, USA: ACM, 1995. v. 24. 163–174 p. ISSN 0163-5808. Disponível em: <<http://doi.acm.org/10.1145/568271.223812>>. Citado 2 vezes nas páginas 49 e 50.
- FODOR, I. K. *A survey of dimension reduction techniques*. [S.l.], 2002. Citado 2 vezes nas páginas 49 e 79.
- FRENKEN, K. et al. Entropy statistics and information theory. *Chapters*, Edward Elgar Publishing, 2007. Citado na página 46.
- GAN, J.; TAO, Y. Dbscan revisited: mis-claim, un-fixability, and approximation. In: *ACM. Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. [S.l.], 2015. p. 519–530. Citado 4 vezes nas páginas 51, 53, 54 e 55.
- GHODSI, A. Dimensionality reduction a short tutorial. *Department of Statistics and Actuarial Science, Univ. of Waterloo, Ontario, Canada*, v. 37, p. 38, 2006. Citado na página 49.
- GUNAWAN, A.; BERG, M. de. A faster algorithm for dbscan. *Master's thesis, Technische University Eindhoven*, 2013. Citado 5 vezes nas páginas 54, 55, 56, 57 e 80.
- GUO, H.; MAO, N.; YUAN, X. Wysiwyg (what you see is what you get) volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, v. 17, n. 12, p. 2106–2114, Dec 2011. ISSN 1077-2626. Citado na página 37.

- GUO, H.; XIAO, H.; YUAN, X. Multi-dimensional transfer function design based on flexible dimension projection embedded in parallel coordinates. In: IEEE. *Pacific Visualization Symposium (PacificVis), 2011 IEEE*. [S.l.], 2011. p. 19–26. Citado 2 vezes nas páginas 39 e 40.
- GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. *Journal of machine learning research*, v. 3, n. Mar, p. 1157–1182, 2003. Citado na página 45.
- HARRIS, M. Optimizing cuda. 2007. Disponível em: <<https://developer.download.nvidia.com/assets/cuda/files/reduction.pdf>>. Citado na página 66.
- HE, T. et al. Generation of transfer functions with stochastic search techniques. In: *Proceedings of the 7th Conference on Visualization '96*. Los Alamitos, CA, USA: IEEE Computer Society Press, 1996. (VIS '96), p. 227–ff. ISBN 0-89791-864-9. Disponível em: <<http://dl.acm.org/citation.cfm?id=244979.245572>>. Citado 2 vezes nas páginas 35 e 36.
- HEESCH, D.; YAVLINSKY, A.; RÜGER, S. Nn k networks and automated annotation for browsing large image collections from the world wide web. In: ACM. *Proceedings of the 14th ACM international conference on Multimedia*. [S.l.], 2006. p. 493–494. Citado na página 49.
- HOAND, S. M. *Non-metric multidimensional scaling (MDS)*. 2008. Citado na página 49.
- INDYK, P.; MOTWANI, R. Approximate nearest neighbors: towards removing the curse of dimensionality. In: ACM. *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. [S.l.], 1998. p. 604–613. Citado 2 vezes nas páginas 17 e 45.
- JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. Data clustering: a review. *ACM computing surveys (CSUR)*, Acm, v. 31, n. 3, p. 264–323, 1999. Citado na página 51.
- KARAMI, A.; JOHANSSON, R. Choosing dbscan parameters automatically using differential evolution. *International Journal of Computer Applications*, Foundation of Computer Science, v. 91, n. 7, 2014. Citado na página 80.
- KAUFMAN, A. Volume visualization. *The visual computer*, Springer, v. 6, n. 1, p. 1–1, 1990. Citado 2 vezes nas páginas 16 e 20.
- KAUFMAN, A. E. Volume visualization. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 28, n. 1, p. 165–167, mar. 1996. ISSN 0360-0300. Disponível em: <<http://doi.acm.org/10.1145/234313.234383>>. Citado 5 vezes nas páginas 16, 17, 20, 21 e 22.
- KIM, H. S. et al. Dimensionality reduction on multi-dimensional transfer functions for multi-channel volume data sets. *Information visualization*, SAGE Publications, v. 9, n. 3, p. 167–180, 2010. Citado na página 39.
- KINDLMANN, G. Transfer functions in direct volume rendering: Design, interface, interaction. *Course notes of ACM SIGGRAPH*, v. 3, 2002. Citado 7 vezes nas páginas 16, 17, 27, 34, 35, 38 e 39.
- KINDLMANN, G.; DURKIN, J. W. Semi-automatic generation of transfer functions for direct volume rendering. In: ACM. *Proceedings of the 1998 IEEE symposium on Volume visualization*. [S.l.], 1998. p. 79–86. Citado 5 vezes nas páginas 17, 26, 27, 38 e 43.

- KINDLMANN, G. et al. Curvature-based transfer functions for direct volume rendering: Methods and applications. In: IEEE. *Visualization, 2003. VIS 2003. IEEE*. [S.l.], 2003. p. 513–520. Citado na página 30.
- KNISS, J.; KINDLMANN, G.; HANSEN, C. Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets. In: IEEE COMPUTER SOCIETY. *Proceedings of the conference on Visualization'01*. [S.l.], 2001. p. 255–262. Citado 3 vezes nas páginas 27, 28 e 29.
- KNISS, J.; KINDLMANN, G.; HANSEN, C. Multidimensional transfer functions for interactive volume rendering. *IEEE Transactions on visualization and computer graphics*, IEEE, v. 8, n. 3, p. 270–285, 2002. Citado 5 vezes nas páginas 26, 29, 33, 34 e 35.
- KNISS, J. et al. Gaussian transfer functions for multi-field volume visualization. In: IEEE COMPUTER SOCIETY. *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*. [S.l.], 2003. p. 65. Citado na página 33.
- KOHONEN, T. The self-organizing map. *Proceedings of the IEEE*, IEEE, v. 78, n. 9, p. 1464–1480, 1990. Citado na página 40.
- KÖNIG, A.; GRÖLLER, E. Mastering transfer function specification by using volumepro technology. In: *Spring Conference on Computer Graphics*. [S.l.: s.n.], 2001. v. 17, n. 4, p. 279–286. Citado 4 vezes nas páginas 16, 26, 34 e 73.
- KPFRS, L. On lines and planes of closest fit to systems of points in space. In: *Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems (SIGMOD)*. [S.l.: s.n.], 1901. Citado 2 vezes nas páginas 40 e 49.
- KRUSKAL, J. B. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, Springer, v. 29, n. 1, p. 1–27, 1964. Citado na página 39.
- KRUSKAL, J. B.; WISH, M. *Multidimensional scaling*. [S.l.]: Sage, 1978. v. 11. Citado 2 vezes nas páginas 40 e 49.
- LACROUTE, P. Real-time volume rendering on shared memory multiprocessors using the shear-warp factorization. In: ACM. *Proceedings of the IEEE symposium on Parallel rendering*. [S.l.], 1995. p. 15–22. Citado na página 43.
- LEVOY, M. Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, v. 8, n. 3, p. 29–37, May 1988. ISSN 0272-1716. Citado 3 vezes nas páginas 21, 23 e 27.
- LJUNG, P. et al. State of the art in transfer functions for direct volume rendering. In: *Proceedings of the Eurographics / IEEE VGTC Conference on Visualization: State of the Art Reports*. Goslar Germany, Germany: Eurographics Association, 2016. (EuroVis '16), p. 669–691. Disponível em: <<https://doi.org/10.1111/cgf.12934>>. Citado 12 vezes nas páginas 16, 17, 18, 25, 27, 28, 29, 34, 36, 40, 41 e 43.
- LORENSEN, W. E.; CLINE, H. E. Marching cubes: A high resolution 3d surface construction algorithm. In: *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY, USA: ACM, 1987. (SIGGRAPH '87), p. 163–169. ISBN 0-89791-227-6. Disponível em: <<http://doi.acm.org/10.1145/37401.37422>>. Citado na página 23.

LUO, B.; WILSON, R. C.; HANCOCK, E. R. Spectral embedding of graphs. *Pattern recognition*, Elsevier, v. 36, n. 10, p. 2213–2230, 2003. Citado na página 49.

MACQUEEN, J. et al. Some methods for classification and analysis of multivariate observations. In: OAKLAND, CA, USA. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. [S.l.], 1967. v. 1, n. 14, p. 281–297. Citado na página 40.

MARKS, J. et al. Design galleries: A general approach to setting parameters for computer graphics and animation. In: *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1997. (SIGGRAPH '97), p. 389–400. ISBN 0-89791-896-7. Disponível em: <<http://dx.doi.org/10.1145/258734.258887>>. Citado na página 36.

MAX, N. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, IEEE, v. 1, n. 2, p. 99–108, 1995. Citado 2 vezes nas páginas 16 e 25.

MCCORMICK, B. H. Visualization in scientific computing. *Computer graphics*, v. 21, n. 6, 1987. Citado na página 20.

MOYA-ANEGÓN, F.; HERRERO-SOLANA, V.; JIMÉNEZ-CONTRERAS, E. A connectionist and multivariate approach to science maps: the som, clustering and mds applied to library and information science research. *Journal of Information Science*, Sage Publications Sage CA: Thousand Oaks, CA, v. 32, n. 1, p. 63–77, 2006. Citado na página 49.

NAM, J. E.; MAURER, M.; MUELLER, K. A high-dimensional feature clustering approach to support knowledge-assisted visualization. *Computers & Graphics*, Elsevier, v. 33, n. 5, p. 607–615, 2009. Citado na página 27.

NEWMAN, T. S.; YI, H. A survey of the marching cubes algorithm. *Computers & Graphics*, v. 30, n. 5, p. 854 – 879, 2006. ISSN 0097-8493. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0097849306001336>>. Citado 2 vezes nas páginas 23 e 24.

NGUYEN, B. P. et al. A clustering-based system to automate transfer function design for medical image visualization. *The Visual Computer*, v. 28, n. 2, p. 181–191, 2012. ISSN 1432-2315. Disponível em: <<http://dx.doi.org/10.1007/s00371-011-0634-3>>. Citado na página 40.

PAIVA, A. C. de; SEIXAS, R. de B.; GATTASS, M. *Introdução à visualização volumétrica*. 1999. Disponível em: <<http://w3.impa.br/~rbs/pdf/mcc0399.pdf>>. Citado 2 vezes nas páginas 20 e 21.

PAULOVICH, F. V.; SILVA, C. T.; NONATO, L. G. Two-phase mapping for projecting massive data sets. *IEEE Transactions on Visualization and Computer Graphics*, IEEE, v. 16, n. 6, p. 1281–1290, 2010. Citado na página 49.

PEARSON, K. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, Taylor & Francis, v. 2, n. 11, p. 559–572, 1901. Citado 2 vezes nas páginas 46 e 81.

PEDREIRA, O.; BRISABOA, N. Spatial selection of sparse pivots for similarity search in metric spaces. *SOFSEM 2007: Theory and Practice of Computer Science*, Springer, p. 434–445, 2007. Citado 2 vezes nas páginas 59 e 67.

- PFISTER, H. et al. The transfer function bake-off. *IEEE Computer Graphics and Applications*, IEEE, v. 21, n. 3, p. 16–22, 2001. Citado 4 vezes nas páginas 16, 27, 34 e 66.
- PINTO, F. d. M.; FREITAS, C. M. Volume visualization and exploration through flexible transfer function design. *Computers & Graphics*, Elsevier, v. 32, n. 4, p. 420–429, 2008. Citado na página 36.
- PINTO, F. de M.; FREITAS, C. M. de S. Design of multi-dimensional transfer functions using dimensional reduction. In: EUROGRAPHICS ASSOCIATION. *Proceedings of the 9th Joint Eurographics/IEEE VGTC conference on Visualization*. [S.l.], 2007. p. 131–138. Citado 4 vezes nas páginas 33, 40, 43 e 66.
- PLATT, J. C. Fastmap, metricmap, and landmark mds are all nystrom algorithms. In: *In Proceedings of 10th International Workshop on Artificial Intelligence and Statistics*. [S.l.: s.n.], 2005. p. 261–268. Citado na página 49.
- PONCIANO, D.; SEEFELDER, M.; MARROQUIM, R. Graph-based interactive volume exploration. *Computers & Graphics*, Elsevier, v. 60, p. 55–65, 2016. Citado na página 66.
- POTTS, S.; MÖLLER, T. Transfer functions on a logarithmic scale for volume rendering. In: *Proceedings of Graphics Interface 2004*. School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada: Canadian Human-Computer Communications Society, 2004. (GI '04), p. 57–63. ISBN 1-56881-227-2. Disponível em: <<http://dl.acm.org/citation.cfm?id=1006058.1006066>>. Citado 2 vezes nas páginas 34 e 35.
- PRADHAN, K.; BARTZ, D.; MUELLER, K. Signaturespace: a multidimensional, exploratory approach for the analysis of volume data. Universität Tübingen, 2005. Citado na página 39.
- PRASSNI, J.-S. et al. Shape-based transfer functions for volume visualization. In: IEEE. *Visualization Symposium (PacificVis), 2010 IEEE Pacific*. [S.l.], 2010. p. 9–16. Citado 2 vezes nas páginas 30 e 31.
- PRAUCHNER, J. L.; FREITAS, C. M.; COMBA, J. L. D. Two-level interaction approach for transfer function specification. In: IEEE. *XVIII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'05)*. [S.l.], 2005. p. 265–272. Citado 2 vezes nas páginas 34 e 36.
- PREIM, B.; BOTHA, C. P. *Visual computing for medicine: theory, algorithms, and applications*. [S.l.]: Newnes, 2013. Citado 4 vezes nas páginas 29, 30, 32 e 34.
- QIAN, M.; ZHAI, C. Robust unsupervised feature selection. In: *IJCAI*. [S.l.: s.n.], 2013. p. 1621–1627. Citado 2 vezes nas páginas 45 e 78.
- ROPINSKI, T. et al. Stroke-based transfer function design. In: *Volume Graphics*. [S.l.: s.n.], 2008. p. 41–48. Citado na página 37.
- ROWEIS, S. T.; SAUL, L. K. Nonlinear dimensionality reduction by locally linear embedding. *science*, American Association for the Advancement of Science, v. 290, n. 5500, p. 2323–2326, 2000. Citado 3 vezes nas páginas 39, 40 e 49.
- SABELLA, P. A rendering algorithm for visualizing 3d scalar fields. *ACM SIGGRAPH computer graphics*, ACM, v. 22, n. 4, p. 51–58, 1988. Citado na página 25.

- SAEYS, Y.; INZA, I.; LARRAÑAGA, P. A review of feature selection techniques in bioinformatics. *bioinformatics*, Oxford University Press, v. 23, n. 19, p. 2507–2517, 2007. Citado na página 17.
- SCHUBERT, E. et al. Dbscan revisited, revisited: Why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, ACM, v. 42, n. 3, p. 19, 2017. Citado 5 vezes nas páginas 52, 54, 55, 67 e 78.
- SEREDA, P. et al. Visualization of boundaries in volumetric data sets using lh histograms. *IEEE Transactions on Visualization and Computer Graphics*, IEEE, v. 12, n. 2, p. 208–218, 2006. Citado na página 38.
- SHANNON, C. À mathematical theory of communication. bell system technology. *Journal*, 1948. Citado 2 vezes nas páginas 46 e 81.
- STEINHAUS, H. Sur la division des corp materiels en parties. *Bull. Acad. Polon. Sci*, v. 1, n. 804, p. 801, 1956. Citado na página 40.
- STOMPEL, A.; LUM, E. B.; MA, K.-L. Visualization of multidimensional, multivariate volume data using hardware-accelerated non-photorealistic rendering techniques. In: *10th Pacific Conference on Computer Graphics and Applications, 2002. Proceedings*. [S.l.: s.n.], 2002. p. 394–402. Citado na página 21.
- TAPPENBECK, A.; PREIM, B.; DICKEN, V. Distance-based transfer function design: Specification methods and applications. In: *Sim Vis*. [S.l.: s.n.], 2006. p. 259–274. Citado 2 vezes nas páginas 29 e 30.
- TARJAN, R. Depth-first search and linear graph algorithms. *SIAM journal on computing*, SIAM, v. 1, n. 2, p. 146–160, 1972. Citado na página 56.
- TENENBAUM, J. B.; SILVA, V. D.; LANGFORD, J. C. A global geometric framework for nonlinear dimensionality reduction. *science*, American Association for the Advancement of Science, v. 290, n. 5500, p. 2319–2323, 2000. Citado 2 vezes nas páginas 40 e 49.
- TENGINAKAI, S.; LEE, J.; MACHIRAJU, R. Salient iso-surface detection with model-independent statistical signatures. In: IEEE COMPUTER SOCIETY. *Proceedings of the conference on Visualization'01*. [S.l.], 2001. p. 231–238. Citado na página 27.
- TORY, M.; MOLLER, T. Human factors in visualization research. *IEEE transactions on visualization and computer graphics*, IEEE, v. 10, n. 1, p. 72–84, 2004. Citado na página 66.
- TUY, H. K.; TUY, L. T. Direct 2-d display of 3-d objects. *IEEE Computer Graphics and Applications*, v. 4, n. 10, p. 29–34, Oct 1984. ISSN 0272-1716. Citado na página 23.
- TZENG, F.-Y.; LUM, E. B.; MA, K.-L. A novel interface for higher-dimensional classification of volume data. In: IEEE COMPUTER SOCIETY. *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*. [S.l.], 2003. p. 66. Citado 2 vezes nas páginas 36 e 37.
- WANG, L.; ZHAO, X.; KAUFMAN, A. Modified dendrogram of high-dimensional feature space for transfer function design. In: NIH PUBLIC ACCESS. *Visualization: proceedings of the... IEEE Conference on Visualization/sponsoring organization, IEEE Computer Society Technical Committee on Computer Graphics in cooperation with Association for Computing*

Machinery/SIGGRAPH. IEEE Con... [S.l.], 2012. v. 18, n. 1, p. 121. Citado 3 vezes nas páginas 40, 41 e 66.

WANG, L.; ZHAO, X.; KAUFMAN, A. E. Modified dendrogram of attribute space for multidimensional transfer function design. *IEEE transactions on visualization and computer graphics*, IEEE, v. 18, n. 1, p. 121–131, 2012. Citado 2 vezes nas páginas 17 e 33.

XU, D.; TIAN, Y. A comprehensive survey of clustering algorithms. *Annals of Data Science*, Springer, v. 2, n. 2, p. 165–193, 2015. Citado na página 51.

ZHAO, X.; KAUFMAN, A. Multi-dimensional reduction and transfer function design using parallel coordinates. In: *Proceedings of the 8th IEEE/EG International Conference on Volume Graphics*. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2010. (VG'10), p. 69–76. ISBN 978-3-905674-23-1. Disponível em: <<http://dx.doi.org/10.2312/VG/VG10/069-076>>. Citado 2 vezes nas páginas 39 e 42.

ZHOU, L.; SCHOTT, M.; HANSEN, C. Transfer function combinations. *Computers & Graphics*, Elsevier, v. 36, n. 6, p. 596–606, 2012. Citado 10 vezes nas páginas 17, 18, 27, 28, 29, 42, 44, 45, 46 e 80.

ZUDILOVA-SEINSTRA, E.; ADRIAANSEN, T.; LIERE, R. v. *Trends in Interactive Visualization: State-of-the-Art Survey*. 1. ed. [S.l.]: Springer Publishing Company, Incorporated, 2008. ISBN 1848002688, 9781848002685. Citado na página 27.

TERMO DE REPRODUÇÃO XEROGRÁFICA

Autorizo a reprodução xerográfica do presente Trabalho de Conclusão, na íntegra ou em partes, para fins de pesquisa.

São José do Rio Preto, 12 / 04 / 18

Rafael S. Santos

Assinatura do autor