

Leonardo Marques Rocha

**Agrupamento de dados por  
Florestas de Caminhos Ótimos**

Tese de Doutorado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Doutor em Engenharia Elétrica. Área de concentração: Engenharia de Computação.

Orientador: Luis G. P. Meloni

Co-orientador: Alexandre A. X. Falcão

Campinas, SP  
2008

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

R582a Rocha, Leonardo Marques  
Agrupamento de dados por florestas de caminhos ótimos  
Leonardo Marques Rocha. – Campinas, SP: [s.n.], 2008.

Orientadores: Luis Geraldo Pedroso Meloni, Alexandre  
Xavier Falcão.

Tese (doutorado) - Universidade Estadual de Campinas,  
Faculdade de Engenharia Elétrica e de Computação.

1. Análise de imagem. 2. Processamento de imagens. 3.  
Banco de dados. 4. Sistemas de recuperação de informação.  
I. Meloni, Luis Geraldo Pedroso. II. Falcão, Alexandre  
Xavier. III. Universidade Estadual de Campinas. Faculdade  
de Engenharia Elétrica e de Computação. IV. Título.

Título em Inglês: Data clustering by optimum-path florest transform

Palavras-chave em Inglês: Image analysis, Image processing, Databases, Information  
retrieval systems

Área de concentração: Telecomunicações e Telemática

Titulação: Doutor em Engenharia Elétrica

Banca Examinadora: Francisco Assis de Oliveira Nascimento, Roberto de Alencar Lotufo,  
Sérgio Shiguemi Furuie, Yuzo Iano

Data da defesa: 17/12/2008

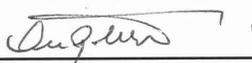
Programa de Pos-Graduação: Engenharia Elétrica

## COMISSÃO JULGADORA - TESE DE DOUTORADO

**Candidato:** Leonardo Marques Rocha

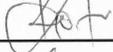
**Data da Defesa:** 17 de dezembro de 2008

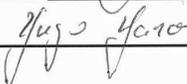
**Título da Tese:** "Agrupamento de Dados por Florestas de Caminhos Ótimos"

Prof. Dr. Luís Geraldo Pedroso Meloni (Presidente): \_\_\_\_\_ 

Prof. Dr. Sérgio Shiguemi Furuie: \_\_\_\_\_ 

Prof. Dr. Francisco Assis de Oliveira Nascimento: \_\_\_\_\_ 

Prof. Dr. Roberto de Alencar Lotufo: \_\_\_\_\_ 

Prof. Dr. Yuzo Iano: \_\_\_\_\_ 

# Resumo

Esta tese propõe uma nova abordagem para agrupamento de dados baseada em floresta de caminhos ótimos. As amostras são tomadas como nós de um grafo, cujos arcos são definidos pela sua relação de adjacência. Os nós são ponderados pela seus valores de densidade de probabilidade (PDF) e uma *função de conexidade* que é maximizada, de forma que cada máximo na PDF torna-se uma raiz de uma árvore de caminhos ótimos (grupo), composto por amostras “mais fortemente conectadas” àquele máximo que a qualquer outra raiz. Máximos irrelevantes (domos) são removidos por reconstrução morfológica da PDF. Uma discussão mostra as vantagens sobre abordagens baseadas no gradiente da PDF e suas relações com a transformada *watershed*. O método é estendido para grande bases de dados e comparado com as abordagens de *mean-shift* e *watershed* para agrupamento de pixels e segmentação interativa, respectivamente. Os experimentos para segmentação de imagens naturais e classificação de substâncias branca e cinza do cérebro mostram resultados rápidos e precisos comparados com métodos no estado-da-arte.

**Palavras-chave:** Agrupamento, Floresta de Caminhos Ótimos, *Watershed*, *Mean-shift*, Transformada Imagem Floresta.

# Abstract

This thesis proposes a novel approach for data clustering based on optimum-path forest. The samples are taken as nodes of a graph, whose arcs are defined by an adjacency relation. The nodes are weighted by their probability density values (pdf) and a *connectivity function* is maximized, such that each maximum of the pdf becomes root of an optimum-path tree (cluster), composed by samples “more strongly connected” to that maximum than to any other root. Irrelevant maxima (domes) are removed by morphological reconstruction of the pdf. A discussion shows the advantages over approaches based on the gradient of the pdf and its relations with the watershed transform. The method is extended for large datasets and compared to mean-shift and watershed approaches for pixel clustering and interactive segmentation, respectively. The experiments for the segmentation of natural images and automatic gray and white matter classification of the brain show fast and accurate results compared to state-of-the-art methods.

**Keywords:** Clustering, Optimum-path Forest, Watershed, Mean-shift, Image Foresting Transform

# Agradecimentos

Aos professores Meloni e Falcão, sou grato pelo suporte e orientação.

Aos colegas do Laboratório RT-DSP.

Aos colegas do Laboratório de Informática Visual — LIV.

Aos demais colegas de pós-graduação, pelas críticas e sugestões.

A minha família pelo apoio durante esta jornada.

Ao CNPq, pelo apoio financeiro.

*Aos meus pais, amigos e colegas*

# Sumário

|  |             |
|--|-------------|
| <b>Página de Rosto</b>                           | <b>i</b>    |
| <b>Resumo/Abstract</b>                           | <b>iv</b>   |
| <b>Agradecimentos</b>                            | <b>v</b>    |
| <b>Dedicatória</b>                               | <b>vi</b>   |
| <b>Sumário</b>                                   | <b>vii</b>  |
| <b>Lista de Figuras</b>                          | <b>ix</b>   |
| <b>Lista de Tabelas</b>                          | <b>xiii</b> |
| <b>Lista de Símbolos</b>                         | <b>xiv</b>  |
| <b>Trabalhos Publicados Pelo Autor</b>           | <b>xv</b>   |
| <b>1 Introdução</b>                              | <b>1</b>    |
| 1.1 Motivação . . . . .                          | 1           |
| 1.2 Objetivos . . . . .                          | 3           |
| 1.3 Proposta e Contribuições . . . . .           | 4           |
| 1.4 Organização da Tese . . . . .                | 5           |
| <b>2 Métodos Relacionados</b>                    | <b>7</b>    |
| 2.1 Técnicas de Agrupamento . . . . .            | 12          |
| 2.1.1 Algoritmos Hierárquicos . . . . .          | 12          |
| 2.1.2 Algoritmos Particionais . . . . .          | 13          |
| 2.2 Outras Abordagens para Agrupamento . . . . . | 20          |
| 2.2.1 Abordagens Evolucionárias . . . . .        | 20          |
| 2.2.2 Redes Neurais Artificiais . . . . .        | 21          |
| 2.2.3 Abordagens Baseadas em Buscas . . . . .    | 21          |
| 2.3 Restrições de Domínio . . . . .              | 22          |
| 2.3.1 Restrição Espacial . . . . .               | 23          |
| 2.4 Grandes Conjuntos de Dados . . . . .         | 24          |

|          |  |           |
|----------|--|-----------|
| 2.5      | Classificação Bayesiana . . . . .                                  | 25        |
| 2.5.1    | Função Discriminante . . . . .                                     | 26        |
| 2.6      | Outras abordagens . . . . .  | 27        |
| <b>3</b> | <b>Transformada Imagem-Floresta</b>                                | <b>28</b> |
| 3.1      | Imagens como Grafos . . . . .                                      | 28        |
| 3.2      | Notações e Definições . . . . .                                    | 29        |
| 3.2.1    | Caminhos Ótimos . . . . .  | 30        |
| 3.3      | Aplicações da IFT . . . . .  | 33        |
| 3.3.1    | Operadores na Superfície da Imagem . . . . .                       | 35        |
| 3.3.2    | Fechamento de Bacias e Domos . . . . .                             | 37        |
| 3.3.3    | A Transformada <i>Watershed</i> . . . . .                          | 38        |
| <b>4</b> | <b>Agrupamento por Floresta de Caminhos Ótimos</b>                 | <b>41</b> |
| 4.1      | Método Proposto . . . . .  | 42        |
| 4.1.1    | Estimação de Densidade . . . . .                                   | 43        |
| 4.1.2    | Estimação do Melhor Grafo de $k$ -vizinhos Mais Próximos . . . . . | 44        |
| 4.1.3    | Zonas de Influência de Máximos . . . . .                           | 46        |
| 4.1.4    | Zonas de Influência de Sementes . . . . .                          | 49        |
| 4.2      | Extensão para Grandes Bases de Dados . . . . .                     | 50        |
| 4.3      | Aplicação em Segmentação de Imagem/Vídeo . . . . .                 | 51        |
| 4.3.1    | Características Multi-Escalas . . . . .                            | 52        |
| 4.3.2    | Redução de Resolução Espacial . . . . .                            | 56        |
| 4.3.3    | Restrição de Conexidade Espacial . . . . .                         | 57        |
| 4.3.4    | Segmentação Interativa de Imagens . . . . .                        | 59        |
| 4.3.5    | Aplicações em Diversos Conjunto de Dados . . . . .                 | 62        |
| <b>5</b> | <b>Análise Comparativa e Resultados Experimentais</b>              | <b>63</b> |
| 5.1      | Análise Comparativa . . . . .                                      | 63        |
| 5.2      | Comparações Experimentais com Outros Métodos . . . . .             | 64        |
| 5.2.1    | Segmentação Não-Supervisionada . . . . .                           | 66        |
| 5.2.2    | Segmentação Supervisionada . . . . .                               | 72        |
| 5.3      | Resultados Experimentais . . . . .                                 | 74        |
| 5.3.1    | Segmentação Automática de Imagens Naturais . . . . .               | 74        |
| 5.3.2    | Classificação de Estruturas do Cérebro . . . . .                   | 77        |
| <b>6</b> | <b>Conclusão</b>   | <b>81</b> |
| 6.1      | Trabalhos Correlatos e Futuros . . . . .                           | 83        |
|          | <b>Referências bibliográficas</b>                                  | <b>85</b> |

# Lista de Figuras

|     |   |    |
|-----|---|----|
| 1.1 | (a) Uma PDF de dois grupos relevantes em um espaço bidimensional (onde amostras mais escuras têm mais valor alto na PDF). Os máximos $A$ e $B$ competem pela amostra $C$ oferecendo-lhe caminhos com mesma força de conectividade. (b) As zonas de influência dos máximos da PDF e (c) as zonas de influência de seus máximos relevantes. . . . .   | 2  |
| 2.1 | (a) Espaço de características com uma concentração de amostras por grupo. É possível identificar um número diferente de grupos dependendo da escala. (b) Conjunto visto com o dobro da escala em (a). (c) Detalhe da zona sombreada em (a) onde um ou mais grupos podem ser definidos dependendo da escala ou métrica de similaridade. (d) Amostras em (c) são ilustradas mais próximas entre si na horizontal devido a uma mudança de métrica. . . . . | 8  |
| 2.2 | Passos gerais do procedimento de agrupamento. . . . .   | 9  |
| 2.3 | Um grupo curvilíneo cujos pontos são aproximadamente distante da origem. Diferentes representações (sistema de coordenadas) podem causar diferentes resultados de saída de algoritmos de agrupamento. . . . .   | 9  |
| 2.4 | Um conjunto de dados forma duas elipses em um espaço bidimensional estão separadas em uma dimensão horizontal, mas misturadas na dimensão vertical. Uma redução de dimensionalidade deve eliminar somente características referentes à dimensão vertical. . . . .   | 10 |
| 2.5 | Taxonomia de diferentes abordagens de agrupamento de dados. No nível superior da hierarquia existe uma distinção entre abordagens hierárquicas e particionais. Alguns aspectos abrangem todas as abordagens, independentemente de sua localização na taxonomia. . . . .   | 13 |
| 2.6 | Exemplo de dendrograma obtido usando o algoritmo de <i>single-link</i> . Polígonos convexos e côncavos são agrupados de acordo com sua semelhança com os restante dos membros dos grupos. . . . .   | 14 |
| 2.7 | (Em um conjunto de dados de círculos e triângulos, com ruído representado por pentágonos, (a) o algoritmo de <i>single-link</i> sofre de uma efeito de encadeamento enquanto (b) o algoritmo de <i>complete-link</i> produz grupos compactos e bem agrupados. . . . .   | 14 |

- 3.1 Um caminho  $\pi_s$  estendido por um arco  $\langle s, t \rangle$ . (b) Um grafo com pesos no nós, onde existem dois máximos com valores 3 e 5. Os nós maiores indicam o conjunto semente  $S$ . (c) Caminhos triviais  $f_{min}(\langle t \rangle)$  para cada nó  $t$ . (d) Floresta de caminhos ótimos  $P$  para  $f_{min}$  e os valores de fim de caminho  $V(t)$ . O caminho ótimo  $P^*(t)$  dado pela Equação 3.8 (linha pontilhada) pode ser obtido percorrendo os predecessores até a raiz  $R(t)$  para cada amostra. . . . . 31
- 3.2 (a) Estrutura de Dial para a fila  $Q$ . (b) A estrutura de dados proposta em [1]. . . . . 34
- 3.3 (a) Um grafo 4-conexo com uma semente  $(2, 2)$  dentro de um objeto sombreado e uma semente  $(4, 0)$  no fundo. (b)-(f) Floresta após 5 iterações, onde a semente interna está conquistando e inserindo pixels de fundo em  $Q$ . (g) Floresta após 12 iterações. A semente externa começa a reconquistar os pixels de fundo. (h) Floresta após 20 iterações. (i) Floresta final após 25 iterações. . . . . 36
- 3.4 Um espaço unidimensional onde a intensidade de  $I(s)$  é representada por barras brancas sobrepostas sobre barras cinzas de  $H(s)$ . (a) Sementes são utilizadas como marcadores binários, onde o custo inicial  $H(s) = I(s)$  nas sementes e  $H(s) = +\infty$ , caso contrário. (b) O resultado da IFT em (a) preenche bacias com valor de  $V(s)$  e duas árvores são formadas com rótulos sólido e rachurado. (c) Sementes são utilizadas em todos os pixels com valores  $H(s) > I(s)$ . (d) Caminhos criados em (c) competem entre si para criar uma única árvore que conquista todos os pixels e preenche três bacias. 38
- 3.5 (a) Imagem de ressonância magnética do pulso. (b) Uma função gradiente de  $I(t)$  e (c) seu fechamento  $I(s) + 1$  (invertidos para ilustração) usando uma adjacência de raio  $h_i = 2.5$ . (d) Função  $L(t)$  é calculada a partir de adjacência de raio  $h_i = 3.5$ . . . 40
- 4.1 (a) Um espaço bidimensional onde amostras são representadas por pontos. (b) O peso de cada amostra dado pela distribuição de densidade de probabilidade  $\rho(s)$  dado pela Equação 4.1, representada pela topologia da superfície 3D e pelo inverso do brilho de sua projeção 2D no plano abaixo. (c) Os grupos são definidos pelos máximos da PDF, cujas zonas de influência são ilustradas na projeção 2D pelas regiões de cores preta e branca. (d) As amostras recebem o rótulo de seus respectivos máximos. . . . 45
- 4.2 (a) Espaço de característica com uma concentração de amostra por grupo. É possível identificar um número diferente de grupos dependendo da escala: (b) quatro, em uma escala maior, ou (c) cinco grupos em uma escala menor. . . . . 45
- 4.3 (a) Barras cinzas ilustram um histograma derivado da PDF  $\rho$  unidimensional com quatro máximos locais. (b) A floresta de caminhos ótimos (setas) sobre  $V$  (barras brancas), indicam as zonas de influência de dois máximos restantes depois da filtragem de  $H(t) = \rho(t) - 2$ . . . . . 48
- 4.4 O interior e exterior do contorno sobre a imagem de entrada nível de cinza indicam dois grupos. Estes resultados são obtidos com o Algoritmo 3,  $d_i = 5$ ,  $h$  criado a partir da eliminação de domos de  $\rho$  com área menor que  $10 \cdot 10^3$  pixels, e vetores de características multi-escala criados por: (a) filtro Laplaciano com 5 escalas, (b) filtro Laplaciano com 15 escalas, (c) ASF por reconstrução com 5 escalas, e (d) ASF por reconstrução com 15 escalas. Neste exemplo, a largura de banda  $d_f$  foi calculada pela Equação 4.1 para o melhor grafo  $k$ -vizinhos mais próximos usando sub-amostragem de 16 para 1. . . . . 53

|      |  |    |
|------|--|----|
| 4.5  | (a-d) Imagens coloridas de cenas naturais apresentadas em nível de cinza. . . . .  | 54 |
| 4.6  | (a-d) Imagens de densidade $\rho$ calculadas nas respectivas imagens das Figuras 4.5a-d usando Equação 4.1, onde um maior o valor de brilho indica um maior valor de densidade na PDF. . . . .   | 54 |
| 4.7  | (a-d) Imagens de gradiente calculadas nas respectivas imagens das Figuras 4.5a-d usando Equação 4.17, onde um menor o valor de brilho indica um maior valor de gradiente. . . . .  | 54 |
| 4.8  | Imagem onde o objeto é a família de ursos. (a-b) Um conjunto semente inicial e a imagem correspondente $\rho(l = 1   s)$ . (c-d) Uma melhor estimaco onde sementes internas e externas tentam separar caractersticas do objeto so adicionadas em $\rho(l = 1   s)$ . . . . .  | 55 |
| 4.9  | (a) A sub-amostragem da imagem a cada 16 pixels, verticalmente e horizontalmente,  ilustrada por marcadores de cor branca. (b) Os valores de densidade so representados por valores de brilho (equalizados) da imagem sub-amostrada em (a) os mximos do sub-grafo gerado so marcados com "x". (c) Os demais pixels da imagem em (a) so rotulados com os rtulos das razes do sub-grafo em (b) que lhes ofereceriam um caminho timo se fizessem parte do sub-grafo. . . . .  | 57 |
| 4.10 | (a) O nmero de mximos na imagem, encontrado com $k = 29$ vizinhos, $d_i = 5$ e $d_f = 1.705083$ ,  representado pelo nmero de regies que segmentam a imagem. (b) Os valores de densidade estimada com restrico espacial so representados por valores de brilho na imagem. (c) Um grande nmero de mximos em (a)  filtrado por uma filtragem por abertura de volume de $30 \cdot 10^3$ pixels. . . . .   | 58 |
| 4.11 | (a) Uma limiarizaco na PDF em 95% do seu valor mximo gera componentes que so rotuladas de acordo com sua conexidade na imagem que servem de sementes para (b) o Algoritmo 4, que une os mximos com mesmos rtulos. (c) Os marcadores de objeto (branco) e fundo (preto) so impostos pelo usurio, mas o processo no  intuitivo. . . . .   | 60 |
| 4.12 | (a) Marcadores de objeto e fundo so utilizados para obter (b) a distribuo de probabilidade condicional do objetos. (c) Pesos para a IFT so obtidos pela combinao linear das magnitudes dos gradientes de brilho e probabilidade condicional do objeto. (d) As parties da imagem obtidas pelo Algoritmo 3 com filtragem e restrico espacial so rotuladas como fundo. (e) Marcadores de objeto unem pela IFT (parcial) as regies em que esto envolvidos. (f) Os marcadores de objeto (branco) e fundo (preto) so impostos pelo usurio onde ocorrem falhas no delineamento por OPF, que usa a IFT com o gradiente em (c) para correo. . . . . | 61 |
| 5.1  | Bases de dados de pontos 2D: (a) cone-torus, (b) saturn, (c) petals, e (d) boat. . . . .   | 64 |
| 5.2  | Bases de dados de pontos 2D: (a) data1, (b) data2, (c) data3, (d) data4 e (d) data5. . . . .   | 64 |
| 5.3  | Exemplos de formas MPEG-7 de classes (a)-(c) peixe e (d)-(f) camelo. . . . .   | 65 |
| 5.4  | (a) A imagem nvel de cinza de um falco em uma rvore. (b) As bordas dos grupos sobre (a) no caso de uma super-segmentaco. (c) O nmero de regies irrelevantes  reduzindo pela escolha de $H$ com base na abertura de volume em $\rho$ . (d) O resultado do algoritmo de <i>mean-shift</i> com a unio de regies. . . . .   | 68 |

|      |  |    |
|------|--|----|
| 5.5  | (a-d) Exemplos obtidos pelo algoritmo de <i>mean-shift</i> com a união de regiões. (e-h) Resultados com o método proposto para as mesmas imagens. . . . .  | 69 |
| 5.6  | (a) Superfície de uma PDF bidimensional, pela qual caminhos até os dois máximos onde domos e vales não são planos. (b) PDF em que os maiores valores correspondem a um platô de um máximo local e os máximos são isolados por vales onde seu valor é zero. . . . .   | 70 |
| 5.7  | (a) Uma fatia de ressonância de um pulso com ossos e veias. (b) A magnitude do gradiente da Equação 4.17 é usado como valores para os arcos no grafo e (c) os valores iniciais $h(s)$ para os nós (invertidos e equalizados para ilustração) são entrada da transformada <i>I níveis de cinza</i> com $d_i = 2.8$ e abertura de volume de $50 \cdot 10^3$ pixels. (d) Os objetos, veias e ossos, são corretamente segmentados com poucos defeitos. (e) O resultado para o Algoritmo 3, com $d_i = 5$ e abertura de volume de $15 \cdot 10^3$ pixels, tem resultado com menos artefatos que (f) o resultado do algoritmo de <i>mean-shift</i> , ambos usando apenas a característica de luminância. . . . . | 71 |
| 5.8  | Resultado de agrupamento usando o Algoritmo 3 e $f_2$ com (a)-(d) $H(t) = \rho(t) - 1$ e (e)-(h) $H$ na abertura de volume de $\rho$ . (i)-(l) Resultados com <i>IFT-watershed</i> de marcadores rotulados [2]. . . . .  | 72 |
| 5.9  | (a)-(d) O usuário seleciona marcadores para juntar regiões e/ou separam dada região. (e)-(h) Resultados das segmentações. (i)-(l) Resultados similares com a transformada <i>IFT-watershed</i> de marcadores rotulados. O envolvimento do usuário pode ser diminuído com a guia visual de (a)-(d). . . . .   | 73 |
| 5.10 | Gráficos dos valores presentes na Tabelas 5.3, 5.4 e 5.5 das respectivas medidas de (a) precisão, (b) revocação e sua (c) média harmônica para o conjunto de treinamento. (d) Média harmônica resultante no conjunto de teste nos eixos de <i>precision</i> e revocação  | 76 |
| 5.11 | (a) Os marcadores mostram voxels de WM e GM voxels com mesmo valor de intensidade 1642. (b) Os erros (destacados em branco) são concentrados nas bordas entre GM e WM. . . . .   | 80 |

# Lista de Tabelas

|     |  |    |
|-----|--|----|
| 5.1 | Código para as melhores combinações de método, similaridade, função e critério de otimalidade na biblioteca CLUTO. Os valores de pureza para estas combinações em seus respectivas bases de dados são listadas na Tabela 5.2. É usada a mesma nomenclatura da biblioteca CLUTO para seus parâmetros. . . . . | 66 |
| 5.2 | As colunas mostram bases de dados, valores de pureza obtidas pelos métodos na CLUTO e OPF, com os respectivos número de classes, códigos de combinação e número mínimo de grupos entre parênteses, indicados na Tabela 5.1 e no texto. . . .   | 67 |
| 5.3 | Medidas de precisão para o conjunto de treinamento de 200 imagens a partir de da variação dos parâmetros $k_{max}$ e da quantidade de pixels utilizados na filtragem de volume em $H$ . . . . .  | 75 |
| 5.4 | Medidas de revocação do conjunto de treinamento de 200 imagens a partir de da variação dos parâmetros $k_{max}$ e $d$ na filtragem de volume em $H$ . . . . .  | 75 |
| 5.5 | Médias harmônicas do conjunto de treinamento de 200 imagens a partir de da variação dos parâmetros $k_{max}$ e da quantidade de pixels utilizados na filtragem de volume em $H$ . . . . .  | 76 |
| 5.6 | Classificação de GM de cérebros sintéticos (fantasmas) com pares de ruído e não-homogeneidade entre parênteses: média e desvio padrão de similaridades <i>Dice</i> em $OPF_1$ , $OPF_2$ , $OPF_2 + Bayes$ e na abordagem de Awate et.al. [3]. . . . .  | 78 |
| 5.7 | Classificação de WM de cérebros sintéticos (fantasmas): média e desvio padrão de similaridades <i>Dice</i> em $OPF_1$ , $OPF_2$ , $OPF_2 + Bayes$ e na abordagem de Awate et.al. [3]. . . . .  | 79 |
| 5.8 | Classificação de GM de cérebros reais (ISBR): média e desvio padrão de similaridades <i>Dice</i> em $OPF_1$ , $OPF_2$ , $OPF_2 + Bayes$ e na abordagem de Awate et.al. [3]. . . .  | 79 |
| 5.9 | Classificação de WM de cérebros reais (ISBR): média e desvio padrão de similaridades <i>Dice</i> em $OPF_1$ , $OPF_2$ , $OPF_2 + Bayes$ e na abordagem de Awate et.al. [3]. . . .  | 79 |

# Lista de Símbolos

- $f(\pi)$  - Função de um caminho  $\pi$
- $\rho$  - Probabilidade *a priori*
- $\vec{v}$  - Vetor de características extraídas da imagem
- $\mathbf{x}$  - Vetor de observação de características em uma base de dados
- $\mathcal{A}$  - Relação de adjacência
- $\mathcal{N}$  - Conjunto de nós do grafo definido no domínio da imagem
- $\mathcal{P}$  - Probabilidade *a posteriori*
- $\mathcal{R}$  - Região de decisão de classificação
- $V(s)$  - Valor de caminho ótimo com término em  $s$
- $H(s)$  - Valor inicial de caminho trivial em  $s$
- $L$  - Mapa de rótulos
- $P$  - Mapa de predecessores
- $R$  - Mapa de raízes
- $S$  - Conjunto de sementes

# Trabalhos Publicados Pelo Autor

1. L. M. Rocha, A. X. Falcão e L. G. P. Meloni. A robust extension of the mean shift algorithm using optimum path forest. Proc. of the 12th Intl. Workshop on Combinatorial Image Analysis, 2008.
2. L. M. Rocha, F.A.M. Cappabianco e A.X. Falcão. Data clustering as an optimum-path forest problem with applications in image analysis. International Journal of Imaging Systems and Technology, 2008.
3. L. M. Rocha, F. A. M. Cappabianco e A. X. Falcão. Data clustering as an optimum-path forest problem with applications in image analysis. Relatório Técnico IC-08-22. Setembro 2008.
4. L. M. Rocha, F. A. M. Cappabianco e A. X. Falcão. Clustering by optimum path forest and its application to automatic GM/WM classification in MR-T1 images of the brain. Relatório Técnico IC-08-02. Janeiro 2008.
5. L. M. Rocha, A. X. Falcão e L. G. P. Meloni. Data clustering based on optimum-path forest and probability density function. Relatório Técnico IC-07-31. Outubro 2007.
6. F. P. G. Bergo, A. X. Falcão, P. A. V. Miranda e L. M. Rocha. Automatic image segmentation by tree pruning. Journal of Mathematical Imaging and Vision, 29(2-3):141-162, Novembro 2007.
7. P. A. V. Miranda, F. P. G. Bergo, L. M. Rocha e A. X. Falcão. Tree-pruning: A new algorithm and its comparative analysis with the watershed transform for automatic image segmentation. XIX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAP), pp 37-44, Outubro 2006.

# Capítulo 1

## Introdução

A identificação de grupos naturais compostos por amostras de um conjunto de dados, denominada em inglês *clustering*, é uma etapa crucial em muitas aplicações de análise de dados. Atributos (observações, medidas ou características) são extraídos das amostras e armazenados em um vetor de atributos por amostra (ponto do  $\mathcal{R}^n$ , para  $n$  atributos). Grupos naturais são caracterizados por amostras com padrões característicos similares, as quais formam altas concentrações de pontos no espaço  $\mathcal{R}^n$  de atributos. A dissimilaridade entre as amostras é medida por uma função de distância (e.g., Euclidiana, Mahalanobis) entre os respectivos vetores de atributos.

Para uma dada aplicação, o agrupamento visa particionar as amostras em um certo número de categorias (classes) sem contar com nenhum conhecimento *a priori* sobre os rótulos das amostras. Por isso, em reconhecimento de padrões, o agrupamento é também denominado aprendizado não-supervisionado. Neste sentido, o agrupamento é um problema mal formulado, pois não existe nenhuma garantia que a rotulação obtida seja a mesma das classes pretendidas (e.g., vários grupos podem estar associados a uma mesma classe). As motivações para as técnicas de agrupamento estão na dificuldade de rotulação das amostras, principalmente em grandes coleções de dados, e no fato que, em muitas aplicações, é possível contar com outras informações, até mesmo a ajuda do usuário, para reconhecer os grupos que compõem cada classe.

### 1.1 Motivação

Entre as várias aplicações, esta tese tem como motivação as aplicações em análise de imagens. Neste contexto, as amostras podem ser imagens de uma coleção em recuperação de imagens por conteúdo (e.g., imagens de peixes em uma base ambiental), objetos segmentados de imagens para classificação de padrões (e.g., espécies de parasitos em imagens de microscopia óptica), ou pixels de uma imagem para segmentação de objetos (e.g., os parasitos). Em cada caso, existem atributos

apropriados que exploram características de forma, cor e textura. Técnicas de agrupamento também podem ser usadas para compressão de imagens, acompanhamento da localização de objetos em um vídeo, etc.

Altas concentrações de pontos no espaço de atributos (Figura 1.1a) formam domos de uma função densidade de probabilidade (*Probability Density Function* — PDF). Estes domos podem ser detectados e separados nos vales entre domos vizinhos. Várias técnicas assumem de forma explícita, ou frequentemente de forma implícita, que os grupos (domos) têm formas conhecidas e/ou podem ser ajustados a funções paramétricas [4, 5, 6, 7]. Dado que as formas podem não ser hiper-elípticas, a premissa mais comum, vários métodos têm evitado premissas sobre a geometria dos grupos [8, 9]. Entre estas abordagens, a técnica *mean-shift* é a mais investigada em análise de imagens [8, 10, 11, 12, 13, 14, 15].

Para cada amostra, a técnica *mean-shift* segue a direção do vetor gradiente da PDF rumo ao máximo mais próximo da amostra. A PDF nunca é calculada explicitamente e cada máximo deveria definir uma “zona de influência” composta pelas amostras que encontram ele. Não é difícil perceber dois problemas críticos desta abordagem. Em regiões onde o gradiente não é bem estimado, por falta de amostras, ou regiões com densidade de probabilidade constante, o método pode falhar na determinação do máximo. Se um máximo consiste de uma região com várias amostras e densidade de probabilidade constante, a zona de influência deste máximo poderá ser dividida em várias zonas de influência, uma para cada de suas amostras. Isto aumenta o número de grupos (*clusters*), o qual é normalmente maior do que o desejado (Figura 1.1b), de maneira que a partição desejada pode requerer a eliminação de máximos (grupos) irrelevantes (Figura 1.1c).

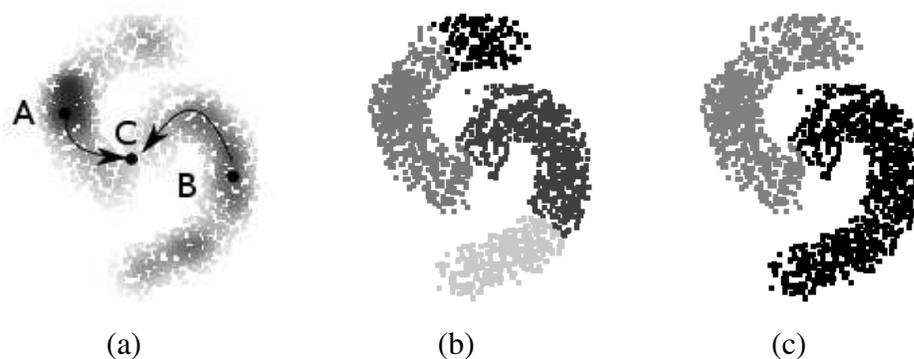


Fig. 1.1: (a) Uma PDF de dois grupos relevantes em um espaço bidimensional (onde amostras mais escuras têm mais valor alto na PDF). Os máximos *A* e *B* competem pela amostra *C* oferecendo-lhe caminhos com mesma força de conexão. (b) As zonas de influência dos máximos da PDF e (c) as zonas de influência de seus máximos relevantes.

## 1.2 Objetivos

Este trabalho teve como objetivos:

1. Prover uma solução robusta e eficiente para o agrupamento por zonas de influência dos máximos da PDF.
2. Prover uma solução mais geral do que a técnica *mean-shift*, permitindo a eliminação de máximos (grupos) irrelevantes.
3. Prover uma solução de particionamento ótimo mais geral em relação à maioria das técnicas de agrupamento. Algumas dessas técnicas são baseadas em critérios *adhoc* locais (e.g., remoção do arco de maior peso em uma árvore espalhada mínima), outras usam critérios globais, mas assumem geometrias conhecidas para os grupos em  $\mathbb{R}^n$  e/ou função paramétrica para a PDF.
4. Prover uma solução robusta para a estimativa da PDF.
5. Avaliação dos resultados no contexto de segmentação de imagens, no qual temos o agrupamento de pixels.

Os objetivos 1 e 2 tiveram como desafio encontrar um algoritmo eficiente que identificasse um único rótulo para todas as amostras de um mesmo máximo da PDF, que não fosse baseado em gradiente, e depois propagasse este rótulo para as demais amostras de forma que as superfícies de decisão fossem formadas nos vales da PDF, ou no meio dos platôs entre zonas de influência de máximos distintos. O objetivo 3 teve como desafio resolver o problema sem assumir geometrias conhecidas para os grupos e função paramétrica para a PDF, gerando uma partição ótima das amostras segundo algum critério global.

A técnica não-paramétrica clássica para estimar a PDF usa janela de Parzen, a qual tem como parâmetro a escolha de uma pequena região do espaço  $\mathbb{R}^n$  em torno da amostra. A forma e dimensões desta região devem levar em conta que os grupos podem ter geometrias arbitrárias e concentrações diferentes de amostras. Os parâmetros da região também variam com a escala. Estes aspectos constituem o desafio do objetivo 4.

Segmentação de imagens constitui um dos maiores problemas em análise de dados. O agrupamento de pixels em duas classes, objeto e fundo, é praticamente impossível sem a ajuda do usuário (segmentação interativa) ou alguma informação específica da aplicação (segmentação automática). O objetivo 5 envolveu, portanto, investigar soluções em ambos paradigmas de segmentação interativa e automática.

## 1.3 Proposta e Contribuições

A tese propõe uma solução baseada em análise combinatória, onde as amostras são nós de um grafo cujos arcos são definidos entre amostras  $k$ -vizinhas mais próximas em  $\mathbb{R}^n$ . Esta solução faz com que o problema de estimar parâmetros reais para a janela de Parzen [16, 17] se transforme em um problema de encontrar um inteiro  $k$  que define os arcos do grafo. A solução proposta é baseada na partição que obtém o corte de peso mínimo do grafo [18].

Uma sequência de amostras forma um caminho  $\pi_t$  no grafo com término em uma dada amostra  $t$  e uma função de valor de caminho  $f(\pi_t)$  calcula a menor densidade de probabilidade ao longo do caminho. Uma partição ótima é obtida pela maximização de  $f(\pi_t)$  para toda amostra  $t$  do conjunto de dados, de forma que cada máximo da PDF define uma zona de influência (árvore de caminhos ótimos) composta pelas amostras mais fortemente conexas a ele do que a qualquer outro máximo. Cada grupo é representado por um máximo e por uma floresta de caminhos ótimos, onde cada árvore tem um rótulo. Por isso, denominados o método Agrupamento por Floresta de Caminhos Ótimos (OPF — *clustering by optimum-path forest*). Um parâmetro de inicialização de  $f(\pi_t)$  permite ainda eliminar grupos representados por domos com baixo volume, área ou altura na PDF.

O método proposto evita os problemas mencionados nos objetivos 1 e 2, encontrados em abordagens baseadas em gradiente, pela identificação inicial de uma amostra por cada máximo relevante da PDF e por conseguinte definição da zona de influência de cada máximo. O método utiliza a robustez do modelamento em grafo [19] para estender o agrupamento do domínio da imagem para o domínio do espaço de características. Uma vantagem do modelo de grafos é a definição de grupos como árvores no grafo, o que permite que grupos tenham formatos que vão além de geometrias conhecidas no espaço  $\mathbb{R}^n$ , como hiper-esferas ou hiper-elipses, definidas por modelos paramétricos. Isto soluciona o problema do objetivo 3, onde são obtidas partições com formatos mais complexos, como ilustrado no exemplo da Figura 1.1).

O número de grupos é encontrado de forma automática pelo algoritmo de agrupamento por OPF. O número de máximos locais relevantes pode ser definido de acordo com a escala um conjunto de amostras pode ser observada. A estimação da escala que minimiza o corte em grafo busca resolver de forma automática o objetivo 4. No entanto, as partições obtidas podem ser posteriormente unidas por critérios baseados no volume, na área das partições ou na altura dos respectivos máximos na PDF, definidos pelo usuário de forma a atender uma configuração de agrupamento mais complexa do que a obtida somente por corte em grafo.

O agrupamento por floresta de caminhos ótimos também foi avaliado com sucesso no contexto de segmentação automática em um banco de imagens de cenas naturais e um banco de imagens de ressonância magnética. No primeiro banco de imagens, o objetivo é delinear objetos automaticamente e comparar com o resultado médio do delineamento humano. No banco de imagens de ressonância

magnética, o objetivo é separar as substâncias cinzenta e branca de cérebros humanos artificiais e reais, com gradual introdução de ruído.

No contexto de segmentação de imagens, a tese propõe o agrupamento de pixels como um pré-processamento que particiona a imagem em algumas regiões de forma que o usuário compõe interativamente um objeto de interesse pela união dessas regiões. Um simples *click do mouse* é necessário nestas regiões. Quando o agrupamento falha e une partes do objeto e do fundo em uma mesma região, o usuário utiliza a técnica de *transformada watershed* de marcadores rotulados (WMT — *Watershed-from-Markers Transform*) para separar os pixels da região.

O método proposto utiliza o arcabouço da Transformada Imagem-Floresta [19] (IFT — *Image Forest Transform*) para a tarefa de agrupamento por OPF e delineamento através da técnica de *watershed* de marcadores [2]. No primeiro caso, uma extensão do algoritmo geral de IFT é utilizada para encontrar os máximos regionais da PDF em um grafo no domínio  $\mathbb{R}^n$ . A IFT é utilizada na implementação do algoritmo de *watershed* para prover uma resposta rápida ao usuário. Neste caso, o pesos dos arcos é dado por uma ponderação de pesos extraídos da imagem e do objeto selecionado pelo usuário através de marcadores de treinamento. Estes experimentos automáticos e interativos constituem o objetivo 5 deste trabalho.

Do ponto de vista teórico, a tese estende a IFT para o espaço de características e revela relações importantes entre a transformada *watershed* de marcadores nível de cinza e o agrupamento por OPF. A IFT tem sido utilizada para reduzir problemas de processamento de imagem em um problema de floresta de caminhos ótimos em um grafo derivado da imagem, pela minimização/maximização de uma função de conectividade. O operador de imagem é calculado para um ou mais atributos da floresta. A função de conectividade usada no espaço de característica é dual à utilizada na transformada IFT-*watershed* de marcadores nível de cinza na domínio da imagem [20, 2], a qual calcula a reconstrução morfológica [21] e a transformada *watershed* na mesma operação. Isto é, os grupos obtidos são equivalentes às regiões de uma PDF filtrada (uma PDF sem os domos irrelevantes), sendo assim uma solução mais geral do que a obtida pelo popular algoritmo de *mean-shift* [8].

## 1.4 Organização da Tese

A estrutura da apresentação dos tópicos seguintes segue organizada em quatro capítulos onde o tema da tese é objetivamente apresentado. O Capítulo 2 apresenta métodos de agrupamento relacionados com este trabalho. Os métodos existentes são inúmeros e sua taxonomia é também bastante diversificada, o que torna difícil entrar em consenso entre diversos trabalhos na literatura. Isto se deve ao fato de muitos destes métodos serem provenientes de área correlatas, mas distintas em seu objetivo, como reconhecimento de padrões, inteligência artificial e processamento de imagem. Os passos co-

muns ao procedimento de agrupamento são brevemente descritos e também os principais métodos da literatura que estão dentro do contexto deste trabalho. Restrições de domínio e extensão para grande base de dados também são introduzidos como paradigmas de algoritmos de agrupamento. Adicionalmente, abordagens semi-supervisionadas também são incluídas, pois fazem parte do treinamento de amostras para problema de segmentação de imagens.

No Capítulo 3, o foco é o arcabouço da IFT, suas notações e definições. O procedimento de preenchimento de bacias é apresentado em conjunto com o algoritmo de *watershed*, pois ambos podem ser calculados em um único algoritmo, onde a diferença entre eles é a saída considerada. Ambos algoritmos podem utilizar marcadores para selecionar possíveis raízes da floresta criada. Para que isto seja possível, é necessária uma função marcadora que seleciona as sementes que efetivamente se desdobrarão em árvores e preencherão bacias selecionadas.

O algoritmo de OPF é descrito no Capítulo 4 como o dual do algoritmo de *watershed* apresentado no Capítulo anterior aplicado na PDF de um grafo entendido para o espaço multidimensional de características com propagação a partir de máximos locais e de marcadores. A primeira versão utiliza o algoritmo de busca de máximos regionais com a rotulação do algoritmo de *watershed* clássico no domínio multidimensional. O segundo algoritmo é baseado no *watershed* de marcadores com rótulos provenientes de um processo *ad hoc* ou interativo. Os aspectos relacionados com a extensão do contexto genérico de grafos para imagens implica em duas abordagens para tornar prático o uso do algoritmo de OPF em grande base de dados. Uma das soluções basea-se na restrição de domínio espacial, e outra em classificação semi-supervisionada em que o usuário fornece informação *a priori* sobre o conjunto de dados.

Os resultados do algoritmo proposto são discutidos no Capítulo 5 através de argumentação teórica e empírica, e comparados com a técnica de *mean-shift*, método em que se baseia o trabalho. A técnica proposta é comparada em termo de suas soluções teóricas para problemas encontrados no algoritmo de *mean-shift*. Resultados empíricos em segmentação de imagens são comparados subjetivamente com *mean-shift* e *watershed* e objetivamente com métodos de segmentação de objetos em um banco de teste de segmentação criado para este propósito.

A conclusão da proposta de tese e trabalhos de tese relacionados, já concluídos ou em andamento, que se beneficiam da teoria proposta são apresentados no Capítulo 6. A trabalhos futuros, tanto no contexto interativo como automático, de segmentação de imagens e aplicações em áreas de compressão e indexação de dados são discutidas sob o ponto de vista da metodologia criada.

# Capítulo 2

## Métodos Relacionados

*Agrupamento* é a organização de uma coleção de amostras em grupos, onde características são extraídas e representadas como por vetores de atributos ou pontos no espaço  $n$ -dimensional de atributos  $\mathbb{R}^n$ . Intuitivamente, amostras dentro de um grupo são mais similares entre si do que qualquer outra amostra pertencente a um grupo diferente. A tarefa do agrupamento é atribuir um mesmo rótulo a amostras que pertencem ao mesmo grupo sem conhecimento inicial de seus rótulos. Cada grupo define uma *zona de influência*, delimitada por uma região  $\mathcal{R}$  no espaço  $\mathbb{R}^n$ . Em agrupamento, esta tarefa parece mal-formulada desde seu início, pois não existe um subconjunto-verdade das amostras que definem as regiões  $\mathcal{R}$ , as quais definem rótulos das amostras. Desta forma, a quantidade de rótulos só é conhecida se provida como entrada do procedimento, pelo usuário ou uma etapa anterior, com uma informação de mais alto nível sobre o conjunto de dados. Adicionalmente, um subconjunto de amostras (sub-amostragem) pode ser utilizado na estimação do provável número de grupos, que está relacionado com a escala de detalhe na qual o conjunto de dados como um todo é observado. Identificar o número de grupos é algo que depende da escala de observação e de medidas de similaridade, como mostra o exemplo da Figura 2.1. No primeiro caso, a escala determina a vizinhança de cada amostra que define o tamanho mínimo de um grupo. A medida de similaridade define a distância entre amostras para o procedimento de agrupamento. Uma forma intuitiva de identificar grupos é através da distribuição de probabilidade (PDF). Em diversos métodos de agrupamento, a sub-amostragem é utilizada na estimação direta e indireta da PDF.

Um procedimento de agrupamento envolve basicamente a sequência de passos ilustrada pela Figura 2.2:

- representação e seleção de características,
- definição de uma medida de proximidade,
- agrupamento,

- abstração dos dados (caso necessário), e
- avaliação da saída (caso necessário).

A *representação de características* refere-se ao número de classes, ao número de características, ao tipo e escala das características, disponíveis ao algoritmo de agrupamento. A natureza da característica pode ser formada de valores contínuos, discretos ou de intervalos de valores, nas escalas racional, nominal ou ordinal. Uma boa representação comumente resulta em um fácil entendimento do grupo, pois uma representação pobre pode resultar em um problema mais complexo de agrupamento, cuja estrutura de dados é difícil ou impossível de distinguir. O exemplo da Figura 2.3 mostra que dependendo da escolha do sistema de coordenadas, o algoritmo de agrupamento pode resultar em diferentes resultados. Se for escolhido o sistema Cartesiano de coordenadas, muitos algoritmos de agrupamento poderão fragmentar o grupo em dois ou mais grupos. Se o sistema de coordenadas polares for utilizado, somente um grupo é facilmente obtido.

Opcionalmente uma *seleção de características* pode ser extraída do conjunto original disponível de forma a identificar o subconjunto mais efetivo. É geralmente valioso isolar apenas as características mais descritivas e discriminatórias de conjunto de entrada, e utilizar essas características exclusivamente em análises subsequentes. Uma outra forma de produzir características mais efetivas é

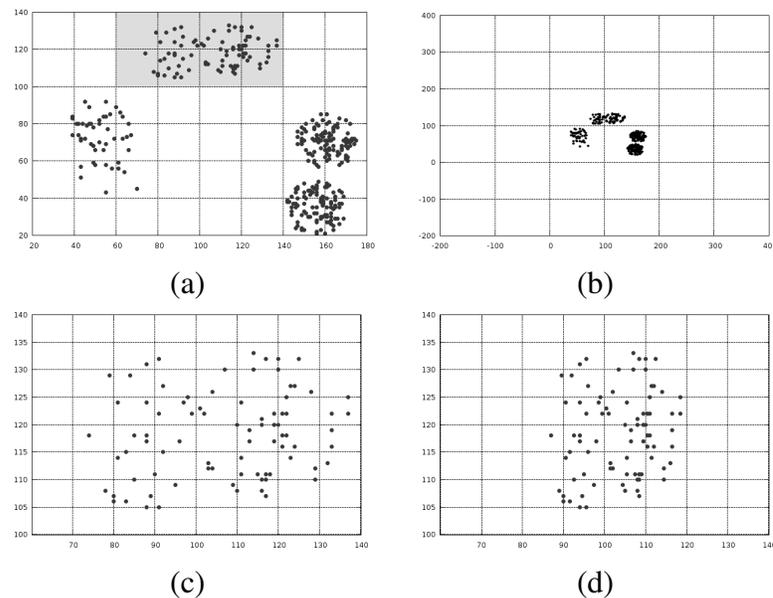


Fig. 2.1: (a) Espaço de características com uma concentração de amostras por grupo. É possível identificar um número diferente de grupos dependendo da escala. (b) Conjunto visto com o dobro da escala em (a). (c) Detalhe da zona sombreada em (a) onde um ou mais grupos podem ser definidos dependendo da escala ou métrica de similaridade. (d) Amostras em (c) são ilustradas mais próximas entre si na horizontal devido a uma mudança de métrica.

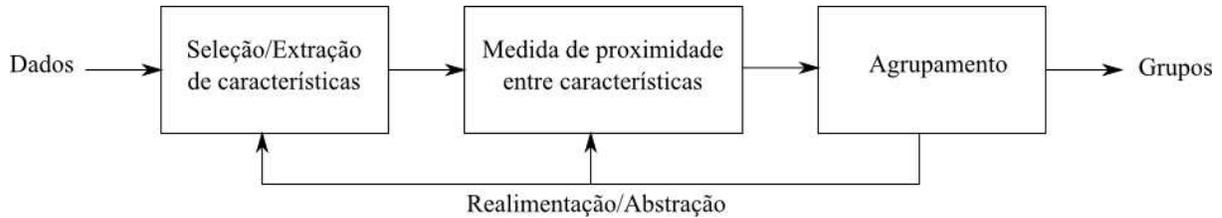


Fig. 2.2: Passos gerais do procedimento de agrupamento.

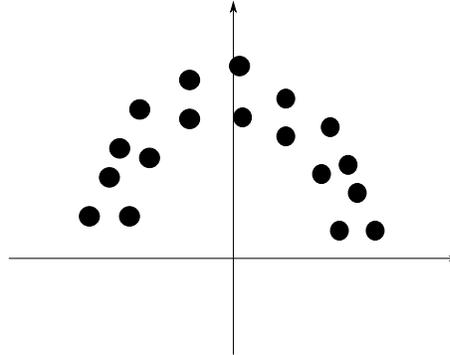


Fig. 2.3: Um grupo curvilíneo cujos pontos são aproximadamente distante da origem. Diferentes representações (sistema de coordenadas) podem causar diferentes resultados de saída de algoritmos de agrupamento.

usar uma *extração de características*, que consiste em uma ou mais transformações no conjunto de entrada.

Técnicas de seleção de características é um tópico bem explorado em reconhecimento estatístico de padrões [22], no entanto, no contexto de agrupamento é um processo que necessita de uma seleção *ad hoc* baseada em tentativa e erro onde a saída de vários subconjuntos de características é analisada por um índice. No entanto, alguns dos processo mais populares — análise de componentes principais [23] (PCA — *principal components analysis*) não depende do rotulamento dos dados e pode ser usado de forma direta. Caso a redução não seja feita de forma correta, grupos distintos podem se misturar em um mesmo subespaço, como o exemplo de um conjunto de dados que forma duas elipses em um espaço bidimensional, como ilustra a Figura 2.4, que estão separadas em uma dimensão horizontal, mas misturadas na dimensão vertical.

A idéia de *proximidade de características* é fundamental para a definição de um grupo. Uma medida de similaridade entre duas características em um mesmo domínio é essencial para os procedimentos de agrupamento e devem ser utilizadas de acordo com a natureza dos dados. Como existem uma variedade de tipos e escalas de dados, esta medida deve ser escolhida com cuidado. No entanto, é mais frequente calcular a dissimilaridade usando uma função de distância definida em pares de características. Uma simples medida como a distância Euclidiana

$$\begin{aligned}
 d_2(\mathbf{x}_i, \mathbf{x}_j) &= \left( \sum_{k=1}^d (x_{i,k} - x_{j,k})^2 \right)^{1/2} \\
 &= \|\mathbf{x}_i - \mathbf{x}_j\|
 \end{aligned}
 \tag{2.1}$$

é comumente usada para refletir a dissimilaridade entre duas características dada uma observação  $\mathbf{x} = (x_1, x_2, \dots, x_n)$ , de uma base de dados. A distância Euclidiana tem um apelo intuitivo e é comumente usada para analisar a distância entre objetos em espaço de 2 ou 3 dimensões. Ela funciona bem quando o conjunto de dados tem grupos compactos ou isolados. A métrica de Minkowski

$$\begin{aligned}
 d_p(\mathbf{x}_i, \mathbf{x}_j) &= \left( \sum_{k=1}^d (x_{i,k} - x_{j,k})^p \right)^{1/p} \\
 &= \|\mathbf{x}_i - \mathbf{x}_j\|.
 \end{aligned}
 \tag{2.2}$$

é uma generalização da distância Euclidiana ( $p = 2$ ). No entanto, a principal desvantagem do uso de métricas de Minkowski é a tendência da característica com maior escala dominar todas as outras. Soluções para este problema incluem a normalização dos valores em um intervalo de variância comum. A correlação linear entre características também podem distorcer medidas de distância, essa distorção pode ser reduzida pelo uso da distância de Mahalanobis

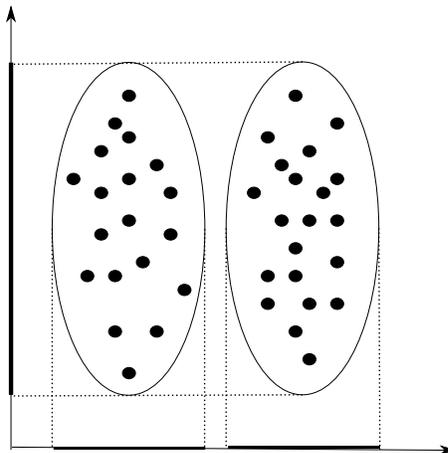


Fig. 2.4: Um conjunto de dados forma duas elipses em um espaço bidimensional estão separadas em uma dimensão horizontal, mas misturadas na dimensão vertical. Uma redução de dimensionalidade deve eliminar somente características referentes à dimensão vertical.

$$d_M(\mathbf{x}_i, \mathbf{x}_j) = (x_i - x_j)^T \Sigma^{-1} (x_i - x_j), \quad (2.3)$$

onde assume-se que os vetores  $x_i$  e  $x_j$  são vetores e  $\Sigma$  é a matriz de co-variância conhecida do processo de geração de padrões. A distância  $d_M$  atribui diferentes pesos para diferentes características baseadas em suas variâncias e correlações lineares entre pares. Assim, é implicitamente assumido que cada classe tem densidades baseadas em distribuições Gaussianas.

O passo de *agrupamento* pode ser realizado de diversas formas. Um agrupamento *hard* particiona o conjunto de dados em grupos que não se sobrepõem. Em contrapartida, o agrupamento *fuzzy* define para cada característica um grau variável de pertinência a cada agrupamento de saída. Algoritmos *hierárquicos* produzem um série de partições aninhadas através de fusão ou divisão de grupos baseados em um critério de similaridade. Algoritmos *particionais* identificam a partição que otimiza um critério de agrupamento. Outras técnicas incluem métodos *probabilísticos* e de teoria de *grafos*.

*Abstração de dados* é o processo de extração de uma representação mais simples e compacta do conjunto de dados após o agrupamento. Isto pode ser necessário durante uma etapa de realimentação, como em procedimentos hierárquicos. A simplicidade é tanto da perspectiva de análise automática, para realizar um procedimento seguinte de forma mais eficiente e compacta, ou do usuário, para que seja fácil de compreensão e intuitiva. O uso do centróide para representar um grupo é a abordagem mais popular e tem bom funcionamento quando a forma dos grupos é compacta e isotrópica. No entanto, quando os grupos são alongados ou anisotrópicos, essa representação não é apropriada. Neste caso, o uso de uma coleção de pontos na borda de um grupo captura sua forma, mas a cardinalidade deste conjunto pode aumentar com a complexidade da forma.

*Análise de validade de grupos* é a avaliação da saída do procedimento de agrupamento. Geralmente a análise utiliza um critério de otimalidade específica, mas critérios são baseados na subjetividade. Uma estrutura de agrupamento é válida se ela não ocorreu por acaso ou como artefato de algoritmo de agrupamento. Existem três tipos de estudo de validação. Um avaliação *externa* compara um estrutura recuperada com uma estrutura *a priori*. Um exame *interno* tenta determinar se a estrutura é intrinsecamente apropriada para os dados. Um teste *relativo* compara duas estruturas e mensura o seus méritos relativos.

## 2.1 Técnicas de Agrupamento

Diferentes abordagens de agrupamento de dados pode ser descritas com a ajuda da hierarquia ilustrada pela Figura 2.5. No nível superior da hierarquia, existe uma distinção entre abordagens hierárquicas e particionais. Métodos hierárquicos produzem uma série aninhada de partições, enquanto métodos particionais produzem apenas um nível de particionamento. No entanto, alguns aspectos abrangem todas as abordagens, independentemente de sua localização na taxonomia. Desta forma, métodos também podem ser considerados aglomerativos ou divisivos, *monothetic* ou *polithetic*, *hard* ou *fuzzy*, determinísticos ou estocásticos, incrementais ou não-incrementais:

- Aglomerativo ou divisivo: este aspecto está relacionado à estrutura do algoritmo ou seu modo de operação. Uma abordagem aglomerativa começa com cada característica em um único grupo e une sucessivamente grupos até que um critério de parada é satisfeito.
- *Monotético* ou *politético*: relaciona-se ao uso sequencial ou simultâneo de de todas as dimensões do vetor de características durante o procedimento de agrupamento. O caso mais comum é o politético, onde todas as distâncias são calculados utilizando todo o vetor. Algoritmos monotéticos usam cada subconjunto do vetor de características de cada vez para tomar uma decisão de agrupamento ou divisão.
- *Hard* ou *fuzzy*: um agrupamento *hard* designa cada amostra a um único grupo durante sua operação e na sua saída. Em contrapartida, um agrupamento *fuzzy* associa um grau de pertinência de vários grupos a cada amostra. Um agrupamento *fuzzy* pode ser convertido em *hard* se grupo com valor mais alto de pertinência é atribuído na saída do processo.
- *Determinístico* ou *estocástico*: este aspecto é mais relevante em métodos baseados em mínimo erro quadrático, onde a procura pelo mínimo da função de erro pode ser feita através de um passo determinístico ou estocástico.
- *Incremental* ou *não-incremental*: esta questão surge quando o conjunto de característica é vasto e existem restrições no tempo de execução ou espaço de memória. O objetivo em tornar um algoritmo incremental é diminuir o número de passagens pelo conjunto de dados, limitar o número de grupos ou reduzir o tamanho das estruturas utilizadas pelo algoritmo.

### 2.1.1 Algoritmos Hierárquicos

Um algoritmo *hierárquico* resulta em uma representação por *dendrograma* (Figura 2.6) de grupos de características aninhados e dos níveis de similaridade em que grupos se juntam. A maioria de

algoritmos hierárquicos são variações de *single-link*, *complete-link* e *minimum-variance* [24]. Os dois primeiros são os mais populares e diferem entre si através da caracterização de [25] similaridade entre dois grupos. No algoritmo de *single-link* a distância entre dois grupos é a *mínima* entre todos os pares de características retirados de dois grupos. A Figura 2.6 mostra o dendrograma correspondente a sete polígonos obtido a partir de um algoritmo de *single-link* de acordo com a semelhança entre suas formas. No algoritmo de *complete-link* a distância entre dois grupos é a *máxima* de todos os pares de distância entre características. Em ambos os casos, os grupos são fundidos para formar um grupo maior baseado no critério de menor distância.

O algoritmo de *complete-link* produz grupos compactos e bem agrupados, mas não consegue extrair grupos concêntricos como o algoritmo de *single-link*. No entanto, este último sofre de uma efeito de encadeamento, cuja tendência é criar grupos que alongados e desordenados. A Figura 2.7 mostra dois grupos separados por uma cadeia de características proveniente de ruído. Os resultados dos algoritmos de *single-link* e *complete-link* são ilustrados pelas regiões dentro das elipses nas Figuras 2.7a e 2.7b, respectivamente.

### 2.1.2 Algoritmos Particionais

Um algoritmo particional obtém uma única partição de dados ao invés de uma estrutura de agrupamento, como o dendrograma produzido pela técnica hierárquica. Métodos particionais têm vantagens em aplicações envolvendo um grande conjunto de dados o qual a construção do dendrograma é computacionalmente proibitivo. Um problema que acompanha um algoritmo particional é a escolha do

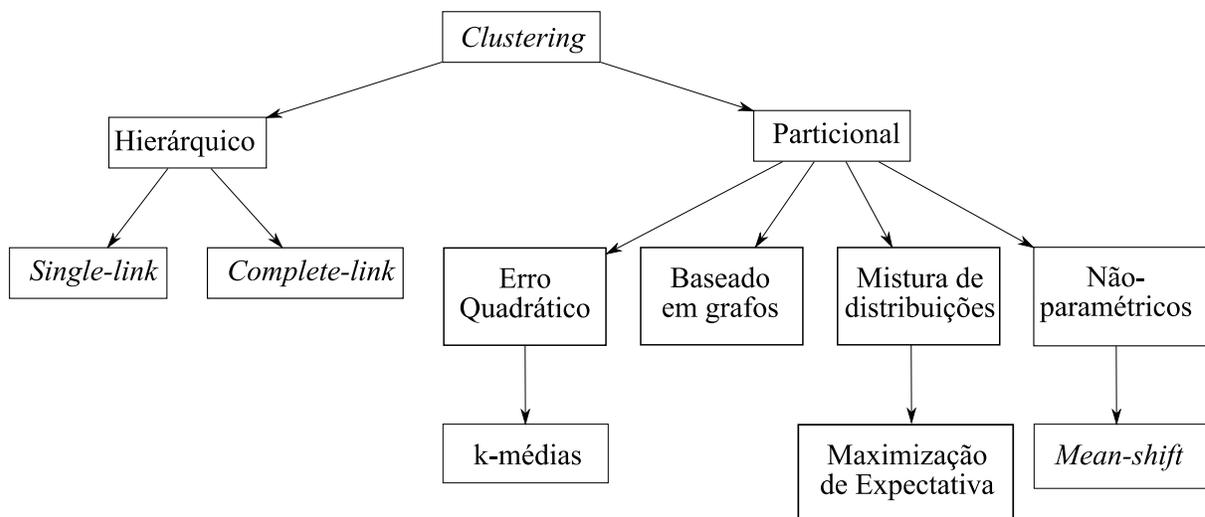


Fig. 2.5: Taxonomia de diferentes abordagens de agrupamento de dados. No nível superior da hierarquia existe uma distinção entre abordagens hierárquicas e particionais. Alguns aspectos abrangem todas as abordagens, independentemente de sua localização na taxonomia.

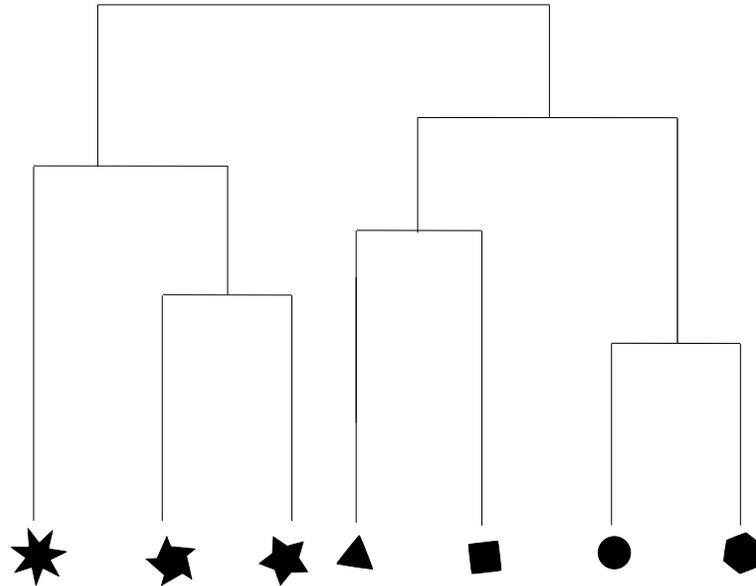


Fig. 2.6: Exemplo de dendrograma obtido usando o algoritmo de *single-link*. Polígonos convexos e côncavos são agrupados de acordo com sua semelhança com os restante dos membros dos grupos.

número de grupos de saída. Os grupos de saída são produzidos pela otimização de uma função objetivo definida localmente (em um subconjunto) ou globalmente (em todo conjunto). Na prática o algoritmo tipicamente é executado múltiplas vezes com diferentes estados iniciais, e a melhor configuração obtida é utilizada como saída de agrupamento.

### Algoritmos de Erro Quadrático Mínimo

A função objetivo mais intuitiva e frequentemente usada em agrupamento particional é o critério de erro quadrático, o qual tende a funcionar bem com grupos compactos e isolados. O erro quadrático para um agrupamento de um conjunto de dados contendo  $k$  grupos é:

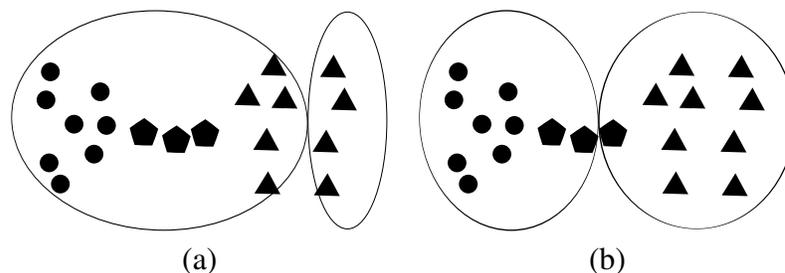


Fig. 2.7: (Em um conjunto de dados de círculos e triângulos, com ruído representado por pentágonos, (a) o algoritmo de *single-link* sofre de uma efeito de encadeamento enquanto (b) o algoritmo de *complete-link* produz grupos compactos e bem agrupados.

$$e^2(\mathbf{x}) = \sum_{j=1}^k \sum_{i=1}^{n_j} \|x_i^{(j)} - x_o^{(j)}\|^2, \quad (2.4)$$

onde  $x_i^{(j)}$  é o  $i$ -ésimo elemento pertencente ao  $j$ -ésimo grupo e  $x_o^{(j)}$  é o centróide do  $j$ -ésimo grupo. O procedimento do método de erros quadráticos mínimos é descrito como se segue:

1. Selecione uma partição de características com um número fixo de grupos e seus centros.
2. Atribua cada elemento ao centro do grupo mais próximo.
3. Para cada grupo, calcule um novo centro do grupo como o centróide do grupo.
4. Repita os passos 2 e 3 até que a convergência seja alcançada, ou seja, a pertinência aos grupos não muda.
5. Faça a união ou separação de grupos baseado em alguma informação heurística, repetindo a passo 2, opcionalmente.

O algoritmo de  $k$ -médias [4] começa com uma partição randômica e atribui características a grupos baseado na similaridade entre o característica e os centro do grupo até que a convergência do critério é alcançada. A convergência ocorre quando não há atribuição de nenhum outra característica proveniente de um grupo para outro, ou o erro quadrático começa a se estabilizar significativamente após um número de iterações. O algoritmo de  $k$ -médias é popular porque ele é fácil de implementar e sua complexidade computacional é linear. Uma desvantagem desse algoritmo é que ele é sensível à seleção da partição inicial e pode convergir a um mínimo local, ao invés de um mínimo global, da função objetivo se a partição inicial não for corretamente escolhida.

Algumas variações do algoritmo de  $k$ -médias tem sido publicadas na literatura, em que é feita uma tentativa de uma seleção inicial da partição para chegar com maior probabilidade ao mínimo global. Uma variação emprega uma técnica de separação e união dos grupos resultantes, como o algoritmo ISODATA [26]. Usando essa variação é possível obter a partição ótima começando de qualquer estado inicial, desde que os valores de limiares para a separação e fusão seja especificados. O procedimento geral de  $k$ -médias é descrito como se segue:

1. Selecione  $k$  centros que coincidem com  $k$  amostras aleatoriamente selecionadas de características ou  $k$  pontos definidos aleatoriamente dentro do hipervolume contendo o conjunto de característica.
2. Atribua cada característica ao centro do grupo mais próximo.

3. Para cada grupo, recalcule o centro do grupo usando as amostras que pertencem ao grupo.
4. Se a convergência não for satisfeita, volte ao passo 2. Critérios de convergência típicos podem ser: nenhuma ou pouca mudança na atribuição de características para novos grupos, ou diminuição mínima do erro quadrático.

### Algoritmos Baseados em Grafos

O algoritmo de agrupamento baseado em grafos mais conhecido constrói uma *árvore espalhada mínima* (*Minimal Spanning Tree* — MST) dos dados. Este algoritmo divide grupos removendo arestas de maior custo da MST. Abordagens hierárquicas também são relacionadas com agrupamento baseado em grafos. Grupos gerados por *single-link* são subgrafos de uma MST dos dados, os quais também são componentes conexos. O resultado de agrupamento de *complete-link* é um sub-grafo completo máximo, que é considerado a definição mais restrita de um grupo e está relacionado com a coloração de grafos.

Grafos de vizinhança também têm sido utilizados em agrupamento, como o grafo de vizinhança relativa (*Relative Neighborhood Graph* - RNG) e o Grafo de Gabriel (*Gabriel Graph* - GG). Estas abordagens são baseadas em regiões de influência [27]. Duas amostras são consideradas vizinhas em um RNG caso nenhuma outra amostra pertença à intersecção das regiões de influência delas, as quais são definidas como sendo discos de raio  $d$  (distância entre elas) centralizados nessas amostras. O Grafo de Gabriel é definido similarmente, porém a região de influência de cada amostra é dada por um disco de diâmetro  $d$ . Tais abordagens possuem como deficiência principal levar em consideração apenas a proximidade entre as amostras, funcionando bem apenas em situações cujos grupos são disjuntos, o que dificilmente ocorre na prática.

Atualmente, o paradigma de achar partições de amostras em um grafo têm sido abordado frequentemente através de medidas de cortes em grafos. O particionamento por corte em grafos usualmente visa atribuir pesos com altos valores para arcos dentro das partições e pesos com valores mais baixos na interface destas. As medidas de corte são definidas por funções objetivos que refletem uma propriedade global nas bordas das partições a partir da distribuição de pesos. Wu e Leahy [18] foram os primeiros a introduzir uma solução para o corte em grafo usando como medida a soma dos pesos de arcos, no entanto sua medida de corte tem uma tendência para bordas pequenas. Outras funções objetivos, como *corte médio* [28], *corte na média* [29], *associação média* [30], *corte normalizado* [31], *corte racional* [32], e *funções de energia* [33, 34, 35] têm sido propostas para contornar este problema.

Seja  $\mathcal{N}_I$  e  $\mathcal{N}_E$  o interior e exterior de um corte  $\mathcal{C}_{IE}$  que consiste de um conjunto de arcos  $(s, t)$ , onde  $s \in I$  e  $t \in E$ . O corte normalizado é definido como:

$$\frac{\mathcal{C}_{IE}}{asso(I) + \mathcal{C}_{IE}} + \frac{\mathcal{C}_{IE}}{asso(E) + \mathcal{C}_{IE}} \quad (2.5)$$

onde

$$\mathcal{C}_{IE} = \sum_{\forall (s,t) | s \in I, t \in E} w(s, t) \quad (2.6)$$

$$asso(I) = \sum_{\forall (s,t) | s \in I, t \in I} w(s, t) \quad (2.7)$$

$$asso(E) = \sum_{\forall (s,t) | s \in E, t \in E} w(s, t). \quad (2.8)$$

O corte médio é definido como

$$\frac{\mathcal{C}_{IE}}{|IE|} \quad (2.9)$$

onde  $|IE|$  é o número de arcos em  $IE$ .

O problema de achar um mínimo na função objetivo através de corte em grafo é um problema NP-difícil para um grafo genérico e muito frequentemente soluções requerem fortes restrições. Soluções heurísticas tem sido propostas em tempo polinomial [36], mas apresentam desempenho computacional pobre e resultados que algumas vezes são bem distantes do que é esperado para o particionamento [37]. Desta forma, verifica-se que mesmo com uma busca reduzida no espaço de soluções que incluem um determinado corte, este nem sempre corresponde ao corte mínimo. Isto sugere que restrições fortes são necessárias na prática. Por exemplo, dois nós terminais, *fonte* e *sorvedouro*, podem ser adicionados ao grafo para representar duas partições de objeto e fundo [33, 34], respectivamente. Adicionalmente à atribuição de pesos a arcos entre os nós originais do grafo, esta abordagem visa atribuir menores pesos a arcos entre o nó fonte e os nós de objeto no grafo original, e entre nó sorvedouro e nós de fundo no grafo original. De forma análoga, pesos maiores são atribuídos a arcos entre nó sorvedouro e nós de objeto no grafo original, e entre nó fonte e nós de fundo no grafo original. Um algoritmo de vazão máxima (*max-flow*), que é análogo ao corte mínimo, da fonte ao sorvedouro [38, 39] é utilizado para o cálculo da partição. Se esse método falha na detecção de uma partição desejada, o usuário pode impor pesos aos arcos entre fonte e destino através da seleção de nós sementes dentro e fora do objeto [33]. O tempo de execução destes algoritmos ainda são polinomiais [34], ou seja, tipicamente  $O(mn^2)$ , onde  $m$  é o número de arcos e  $n$  é o número de nós. Uma solução semi-supervisionada, em tempo linear, apresentada por [40] calcula o conjunto de sementes dentro do objeto, onde a ordem de propagação de cada nó é proporcional ao custo do caminho ótimo no grafo a partir da semente até o nó. O particionamento é feito através das árvores que contém uma ordem de propagação menor, em que a interface entre árvores definem um possível corte. A medida de corte é calculada durante a propagação e a partição obtida pela seleção de um nó com medida de

corte mínimo e, por conseguinte, do conjunto de todos os nós a longo do seu respectivo corte.

A função de energia proposta por [40], semelhante a de [33] e consistente com a formulação geral em [35], é dada por

$$\lambda \left( \sum_{\forall s \in I} (1 - R_o(s)) + \sum_{\forall t \in E} (1 - R_f(t)) \right) + C_{IE} \quad (2.10)$$

onde  $R_o$  e  $R_f$  são as partições calculadas pela Equação 2.11 usando sementes de treinamento dentro do objeto e no fundo, respectivamente; e  $\lambda > 0$  representa a importância do primeiro termo com respeito ao segundo (fator de normalização). A Equação

$$R_s(t) = \exp \left( -\frac{1}{2d} (\mathbf{x}_t - \mu_s)^t \Sigma_s^{-1} (\mathbf{x}_t - \mu_s) \right) \quad (2.11)$$

calcula a pertinência de cada nó  $t$  no grafo à região de  $R_s(t)$  onde as matrizes de média e covariância,  $\mu_s$  e  $\Sigma_s$ , são estimadas por um conjunto de treinamento que inclui o nó  $s$ .

### Algoritmos Baseado em Misturas

Métodos não-paramétricos supõem que as características que serão agrupados são derivados de uma ou mais distribuições. O objetivo é identificar na mistura de distribuições, os parâmetros de cada distribuição. Grande parte dos trabalhos nesta área assumem que componentes individuais da mistura são distribuições Gaussianas. O procedimento tradicional aborda esse problema obtendo iterativamente a estimativa de máxima verossimilhança do vetor de parâmetros das componentes de densidade.

Mais recentemente, o algoritmo de maximização da esperança (*Expectation Maximization* — EM) tem sido aplicado em problemas de estimação de parâmetro. EM é um algoritmo de propósito geral de verossimilhança máxima para problema de dados perdidos [5]. No arcabouço de EM, os parâmetros das componentes de densidade são desconhecidos, como são os parâmetros da mistura de parâmetros, e eles são estimados a partir das características. O procedimento de EM começa com uma estimação inicial do vetor de parâmetros e iterativamente recalcula a medida de verossimilhança a partir de componentes da mistura e compara contra a densidade produzida pelo vetor de parâmetros. Os componentes recalculados da mistura atualizam a estimativa dos parâmetros.

### Algoritmos Não-Paramétricos

Técnicas não paramétricas para agrupamento têm sido desenvolvidas baseadas na estimação de densidade de grupos. Inspiradas pela abordagem da janela de Parzen para estimação da densidade livre de parâmetros, este procedimento procura por intervalos de valores que possuem um grande valor no histograma multidimensional para o conjunto de características de entrada.

*Mean-shift* é um algoritmo não-paramétrico que não requer conhecimento *a priori* do número de grupo e não se limita à forma dos grupos. O algoritmo de *mean-shift* foi proposto inicialmente por Fukugana e Hostetlerin [41] e ficou por muito tempo esquecido até o trabalho de Cheng [8]. A idéia do algoritmo de *mean-shift* é atribuir para cada amostra, o rótulo de um máximo na PDF identificado pelo deslocamento na direção crescente desta. Dado um conjunto de pontos  $x_i, i = 1, \dots, k$  em um espaço  $n$ -dimensional  $\mathfrak{R}^n$  do conjunto de dados, uma estimativa de densidade multivariável através de uma janela de Parzen [16, 17] pode ser obtida por

$$\rho(\mathbf{x}) = \frac{1}{kd^n} \sum_{i=1}^k K\left(\frac{x - x_i}{d}\right). \quad (2.12)$$

onde  $K(\cdot)$  é a janela ou função de kernel e  $d$  é a largura da janela ou o tamanho do kernel. Tradicionalmente, assume-se kernel radialmente simétrico e com  $\int K(x)dx = 1$ . Uma escolha bastante comum é

$$K(\mathbf{x}) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\|\mathbf{x}\|^2}{2}\right). \quad (2.13)$$

As modas da função de densidade estão localizada nos zeros da função de gradiente  $\nabla\rho(\mathbf{x}) = 0$ . O gradiente do estimador de densidade é:

$$\nabla\rho(\mathbf{x}) = \frac{2}{kd^{n+2}} \sum_{i=1}^k (x - x_i) K'\left(\left\|\frac{x - x_i}{d}\right\|^2\right) \quad (2.14)$$

$$\nabla\rho(\mathbf{x}) = \frac{2}{kd^{n+2}} \left[ \sum_{i=1}^k K'\left(\frac{x - x_i}{d}\right)^2 \right] \left[ \frac{\sum_{i=1}^k x_i K'\left(\left\|\frac{x - x_i}{d}\right\|^2\right)}{\sum_{i=1}^k K'\left(\left\|\frac{x - x_i}{d}\right\|^2\right)} - x \right], \quad (2.15)$$

onde  $K'(\cdot)$  é a função de kernel derivada de  $K(\cdot)$ . O primeiro termo da fatoração da Equação 2.14 é proporcional à estimação de densidade em  $x$  calculada com a derivada da função de kernel e o segundo termo:

$$m(\mathbf{x}) = \frac{\sum_{i=1}^k (x_i) K' \left( \left\| \frac{x-x_i}{d} \right\|^2 \right)}{\sum_{i=1}^k K' \left( \left\| \frac{x-x_i}{d} \right\|^2 \right)} - \mathbf{x} \quad (2.16)$$

é o vetor de *mean-shift*. Este vetor sempre aponta em direção do máximo aumento de densidade. O procedimento de *mean-shift* é obtido pelo cálculo sucessivo do vetor  $m(x)$  e translação da janela por este vetor, até a convergência em um ponto onde o gradiente da função de densidade é zero. Todas as amostras cujos deslocamentos convergem para uma mesma moda definem um zona de influência desta associada a um mesmo grupo. O número de partições é desconhecido, mas pode ser imposto pela união de regiões representadas por cada moda. O processo de união pode ser um procedimento sucessivo de *mean-shift* de modas previamente definidas [11] ou baseado em outra heurística [42]. Por exemplo, o agrupamento de pixels em uma imagem geralmente é um passo de pré-processamento para simplificar regiões complexas da imagem, através de características como cor e luminância, e é desejável remover pequenas regiões pela união das regiões conexas mais similares.

## 2.2 Outras Abordagens para Agrupamento

### 2.2.1 Abordagens Evolucionárias

Abordagens evolucionárias são motivadas pela evolução natural e fazem uso de operadores evolucionários e de uma população de soluções para obter a partição globalmente ótima do conjunto de dados. Soluções candidatas para o problema de agrupamento são codificadas como cromossomos. Os operadores evolucionários mais comuns são: seleção, recombinação e mutação. Cada operador transforma um ou mais cromossomos de entrada em um ou mais cromossomos de saída. Uma função de adaptação (*fitness*) é avaliada em um cromossomo determina sua capacidade de sobreviver em uma nova geração. Abaixo segue o algoritmo evolucionário para agrupamento.

1. Escolha uma população aleatória de soluções. Cada solução corresponde a uma partição do conjunto de dados. Associe um valor de adaptação a cada solução. Tipicamente, este valor é inversamente proporcional ao erro quadrático mínimo.
2. Use os operadores de seleção, recombinação e mutação para gerar a próxima população de soluções. Avalie os valores de soluções.
3. Repita o passo 2 até que uma condição de parada é satisfeita.

As técnicas evolucionárias mais conhecidas são Algoritmos Genéticos (AG), Estratégias de Evolução (EE) e Programação Evolutiva (PE). Dentre as três abordagens, AG têm sido mais utilizado em

agrupamento.

### 2.2.2 Redes Neurais Artificiais

Redes neurais artificiais (*Artificial Neural Networks* — ANN) são motivadas pelas redes neurais biológicas e têm sido utilizadas extensivamente nas últimas três décadas tanto para classificação como agrupamento. Algumas das características que tornam ANN importante em agrupamento de padrões são:

- ANN processam vetores numéricos e requerem que padrões sejam representados usando apenas características quantitativas;
- Possuem arquitetura inerentemente paralela e distribuída;
- Aprendem seus pesos de interconexão adaptativamente, mais especificamente, podem atuar como normalizadores de padrões e seletores de características com a seleção apropriada de pesos.

Redes neurais competitivas (*winner-takes-all*) são comumente utilizadas para agrupar dados. Em aprendizado competitivo, características similares são agrupadas pela rede baseada na correlação dos dados. Exemplos bem conhecidos de ANN utilizados em agrupamento incluem Aprendizado de Quantização de Vetores de Kohonen (AQV), Mapa Auto-Organizado (MAO) e modelos de Teoria de Ressonância Adaptativa (TRA). As arquiteturas para estas ANN são simples: são de única camada. Características são apresentadas na entrada e são associadas com nós de saída. Os pesos ficam entre os nós de entrada e saída, onde são iterativamente modificados por aprendizado até que o critério de parado é atingido. O procedimento de atualização dos pesos são similares com aqueles encontrados em algumas técnicas de agrupamento. Exemplos desta relação pode ser encontradas entre algoritmos de  $k$ -médias e AQV. A habilidade de aprender novos dados é denominada plasticidade, que é associado com a estabilidade do sistema de aprendizado. Um sistema é dito estável se nenhuma característica no conjunto de treinamento muda de categoria após um número finito de iterações de aprendizado.

### 2.2.3 Abordagens Baseadas em Buscas

Técnicas de busca são utilizadas para obter o valor ótimo de uma função de critério. Estas abordagens são divididas em buscas determinísticas e estocásticas, dependendo da otimalidade do particionamento. Buscas determinísticas garantem partições ótimas, enquanto buscas estocásticas geram um resultado próximo do ótimo, mas garantem convergência assintótica para a partição ótima como,

por exemplo, técnicas evolucionárias. Técnicas determinísticas são tipicamente abordagens gulosas, enquanto estocásticas permitem perturbações na soluções em direções não-ótimas localmente. A abordagens como Busca Tabu (BT) e Anelamento Simulado (AS) possuem uma probabilidade de aceitação de uma solução. Busca Tabu iterativamente modifica a solução, através de uma busca dentro da vizinhança desta até o critério de parada. Em AS, o parâmetro de aceitação é chamado temperatura, que decresce com a iteração do algoritmo. Desta forma, iterações iniciais têm uma mobilidade maior de busca, que é reduzida gradualmente ao longo da execução do algoritmo.

## 2.3 Restrições de Domínio

A natureza da tarefa de agrupamento é subjetiva, onde um mesmo conjunto de dados pode ser particionado diferentemente de acordo com seu objetivo. Por exemplo, considere um conjunto de dados formado pelos elementos *baleia*, *elefante*, e *atum*. *Baleia* e *elefante* estão no grupo dos mamíferos, no entanto o usuário está interessado em particioná-los baseado no conceito de ambiente natural. A *baleia* e *atum* são animais aquáticos e juntos formam um grupo. Tipicamente essa subjetividade é incorporada no critério de agrupamento através do conhecimento do domínio.

Cada algoritmo de agrupamento usa algum tipo de conhecimento, seja implícito ou explícito. O conhecimento implícito tem sua função desempenhada em passos do processo de agrupamento. Durante a representação de padrões, é utilizado um conhecimento anterior (*a priori*) para selecionar e codificar características. A escolha de medida de distância de Mahalanobis ao invés da distância Euclidiana, por exemplo, é feita para obter grupos hiperelípticos. A escolha de um esquema de agrupamento através do conhecimento da forma dos grupos como, por exemplo, o uso do algoritmo de *k*-médias para obter grupos hiperesféricos.

Também é possível usar explicitamente o conhecimento do domínio para restringir ou guiar o processo de agrupamento. Vários algoritmos utilizam esse esquema especializado em diversas aplicações. Os conceitos relacionados com o domínio podem ter vários papéis durante o processo e uma enorme variedade de escolhas são disponíveis. Em um extremo, o conteúdo do domínio pode ser utilizado como característica adicional e o restante do processo permanece inalterado. Em outro extremo, os conceitos de domínio são utilizados para confirmar ou vetar uma decisão obtida independentemente por um algoritmo de agrupamento, ou afetar o cálculo de distância. Tal incorporação de conhecimento consiste principalmente de abordagens *ad hoc*.

### 2.3.1 Restrição Espacial

O algoritmo de segmentação de pixels por *mean-shift* proposta Comaniciu e Meer [10] é baseada na estimação implícita de PDF restrita ao domínio da imagem. *Mean-shift* foi utilizado para segmentação de imagens e depois entendido para outras aplicações, como rastreamento de objetos em vídeo[12, 11, 15]. Muita pesquisa foi desenvolvida a partir desse algoritmo com o intuito de melhorar resultados de segmentação usando janelas adaptativas [43, 44] ou reduzindo o número de deslocamentos usando um abordagem hierárquica [11].

O gradiente é calculado dentro de uma janela no domínio multidimensional composta por suas projeções no domínio espacial e no espaço de características. Na literatura de reconhecimento de padrões, a técnica não-paramétrica da janela de Parzen, vista na Seção 2.1.2, é utilizada para estimação da PDF.

#### Estimação do Gradiente de Densidade

A janela de Parzen definida pelo kernel  $K$ , que na prática tem um suporte finito limitado por sua largura de banda  $d$ , é utilizada para estimar a função de densidade  $\rho(\mathbf{x})$  dada pela Equação 2.12. No entanto, o algoritmo de *mean-shift* utiliza o gradiente da PDF, calculado pela derivada da função  $f$  que define os pesos da janela. Desta forma, é utilizado um procedimento iterativo de busca da moda da distribuição, em que é feito como o sucessivo deslocamento da janela na direção do maior gradiente na PDF até uma posição de convergência. A posição de convergência equivale a um máximo local de densidade na PDF.

Para segmentação de imagens, o vetor multidimensional é composto por dois domínios com naturezas diferentes: *domínio espacial* e *domínio de características*. O domínio espacial é responsável pela conectividade entre pixels na imagem quando a densidade é estimada. Caso contrário, pixels distantes entre si no domínio espacial mas com características similares estariam dentro da janela. O características espectrais devem ser originadas de outros domínios, como cor, brilho, etc. Equação 2.17 mostra a distribuição de pesos de um kernel multivariável fatorado em um produto de dois kernels simétricos  $K_s$  e  $K_f$ , para os domínios espacial e de características, respectivamente.

$$K_{d_s, d_f}(\mathbf{x}) = \frac{1}{(d_s)^2 (d_f)^{n-2}} K_s \left( \left\| \frac{\mathbf{x}_s}{d_s} \right\|^2 \right) K_f \left( \left\| \frac{\mathbf{x}_f}{d_f} \right\|^2 \right) \quad (2.17)$$

Dado o que função do kernel  $K$  é limitada por sua largura de banda, ou seja, é zero para amostras fora da banda, o cálculo de sua densidade é restrito inicialmente ao domínio espacial. A seguir, cada amostra dentro do alcance do raio de  $d_f$  é utilizado para o cálculo de densidade. O cálculo de densidade é então dado por

$$\rho(\mathbf{x}) = \frac{1}{n(d_s)^2(d_f)^{n-2}} \sum_{i=1}^k K_s \left( \left\| \frac{x_s - x_{s,i}}{d_s} \right\|^2 \right) K_f \left( \left\| \frac{x_f - x_{f,i}}{d_f} \right\|^2 \right), \quad (2.18)$$

onde  $k$  é o número de amostras dentro dos kernels  $K_s$  e  $K_f$ .

As equações de gradiente (Equação 2.14) e do vetor do vetor de deslocamento são computados com base na Equação 2.18.

## 2.4 Grandes Conjuntos de Dados

Existem diversas aplicações onde é necessário agrupar uma grande coleção de padrões. A definição de “grande” tem variado com as mudanças tecnológicas, como memória e tempo de processamento. Na década de 1960, grande significava algumas milhares de características; na atualidade, existem aplicações onde milhões de características de alta dimensionalidade têm que ser agrupadas. Por exemplo, para segmentar uma imagem de tamanho de  $500 \times 500$  pixels, o número de pixels a serem agrupados é  $250 \cdot 10^3$ . Em recuperação de documentos e filtragem de informação, milhões de padrões com dimensionalidade superior a 100 são necessárias.

O algoritmo de  $k$ -médias tem sido o método mais popular para agrupar grande conjuntos de dados devido:

- Sua complexidade de tempo é  $O(nkl)$ , onde  $n$  é o número de características,  $k$  é o número de grupos, e  $l$  é o número de iterações necessárias para a convergência do algoritmo.
- Sua complexidade de espaço é  $O(k+n)$ . Espaço adicional é necessário para armazenar a matriz da dados, mas é possível armazená-la em uma memória secundária e acessá-la de acordo com a característica. No entanto, devido a natureza iterativa do algoritmo, esse processo pode aumentar enormemente o tempo devido ao longo tempo de acesso.
- É independente de ordenamento; para um conjunto de inicial de sementes de centros de grupos, uma mesma partição é gerada invariavelmente da ordem que são apresentadas ao algoritmo.

No entanto, o algoritmo de  $k$ -médias é sensível à seleção inicial de sementes e, no melhor caso, só produz grupos hiperesféricos. Algoritmos hierárquicos são mais versáteis, mas possuem as seguintes desvantagens:

- Sua complexidade de tempo em algoritmos aglomerativos é  $O(n^2 \log n)$ , onde  $n$  é o número de elementos de entrada.

- Na abordagem aglomerativa, sua complexidade de espaço é  $O(n^2)$ . Isto se deve ao armazenamento de uma matriz de similaridade de tamanho  $n \times n$

Uma solução possível para o problema de agrupamento em grande conjuntos de dados com um sacrifício marginal na versatilidade dos grupos é implementar variações de algoritmos mais eficientes. Uma abordagem híbrida onde um conjunto de pontos é escolhido por  $k$ -médias e o restante dos dados são associados a um ou mais amostras de referência ou grupos. Árvores espalhadas mínimas são obtidas para cada grupo de pontos separadamente e são aglutinadas para formar uma aproximação da MST global. Essa abordagem calcula apenas uma fração de todos os possíveis pares de amostras. Em algoritmos incrementais, é assumido que é possível considerar uma característica de cada vez e atribuí-las aos grupos existentes. Assim, o novo dado não afeta o grupos existentes. Uma descrição de alto nível de um algoritmo incremental típico é feita a seguir:

1. Atribua o primeiro ítem de dados a um grupo.
2. Considere o próximo ítem de dados e atribua esse ítem a um dos grupos existentes ou atribua a novo grupo. Essa atribuição é feita baseada em algum critério, como a distância ao novo ítem ou ao centro do grupo.
3. Repita o passo 2 até que todos ítems sejam processados.

## 2.5 Classificação Bayesiana

A principal abordagem estatística para reconhecimento de padrões foi proposta por Bayes através de uma classificação Bayesiana. Normalmente supervisionado, um classificador Bayesiano pode ser construído a partir das probabilidades *a posteriori* de ocorrência de cada classe  $l_j$ ,  $j = 1, 2, \dots, c$ . No contexto de agrupamento, ela é aplicada aos rótulos associados ao conjunto de treinamento após o aprendizado não supervisionado deste. O restante das amostras são classificadas pelas probabilidades *a posteriori*  $\mathcal{P}(l_j | \mathbf{x})$ , expressas pela *Regra de Bayes*:

$$\mathcal{P}(l_j | \mathbf{x}) = \frac{\mathcal{P}(\mathbf{x} | l_j)\rho(l_j)}{\rho(\mathbf{x})} \quad (2.19)$$

onde  $\mathcal{P}(\mathbf{x} | l_j)$  é a densidade de probabilidade condicionada à classe  $l_j$ ,  $\rho(l_j)$  é a probabilidade *a priori* de  $l_j$ , e  $\rho(\mathbf{x})$  é a densidade de probabilidade misturada. Ou seja,

$$\rho(\mathbf{x}) = \sum_{i=1}^c \mathcal{P}(\mathbf{x} | l_i)\rho(l_i) \quad (2.20)$$

é o histograma normalizado de  $\mathbf{x}$  com dimensão  $n$  (*scatterogram*) e  $\mathcal{P}(\mathbf{x} | l_i)$  é o histograma normalizado de  $\mathbf{x}$  dentro da classe  $l_j$ . Um classificador Bayesiano deve, portanto, escolher a classe  $l_j$  cuja  $\mathcal{P}(l_j | \mathbf{x}) > \rho(l_i | \mathbf{x})$  para todo  $i \neq j, i, j = 1, 2, \dots, c$ . Esta escolha se baseia na probabilidade mínima de erro, o que torna o classificador de Bayes ótimo.

### 2.5.1 Função Discriminante

Seja  $\mathcal{R}_i$  uma região de decisão que satisfaz a desigualdade  $\mathcal{P}(l_j | \mathbf{x}) < \mathcal{P}(l_i | \mathbf{x})$  para uma classe  $l_i, i \neq j$ . A probabilidade de acerto  $P_{acerto}$  do classificador Bayesiano é dada por

$$P_{acerto} = \sum_{i=1}^c \mathcal{P}(\mathbf{x} \in \mathcal{R}_i, l_i) \quad (2.21)$$

$$P_{acerto} = \sum_{i=1}^c \mathcal{P}(\mathbf{x} \in \mathcal{R}_i | l_i) \rho(l_i) \quad (2.22)$$

$$P_{acerto} = \sum_{i=1}^c \int \mathcal{P}(\mathbf{x} | l_i) \rho(l_i) d\mathbf{x} \quad (2.23)$$

Dado um modelo canônico de um classificador em que o máximo valor de uma função  $g_i(\mathbf{x})$ ,  $i = 1, 2, \dots, c$  resulta na classificação da classe  $l_i$ , a função discriminante do classificador de Bayes é  $g_i(l_i | \mathbf{x})$ . No entanto, qualquer função  $\phi(g_i(\mathbf{x}))$  monotonicamente crescente pode substituir  $g_i(\mathbf{x})$  sem alterar os resultados. Em particular na modificação de  $\phi'$  para  $g_i(\mathbf{x})$ .

$$\begin{aligned} g_i(\mathbf{x}) &= \mathcal{P}(l_i | \mathbf{x}) \\ g_i(\mathbf{x}) &= \frac{\mathcal{P}(\mathbf{x} | l_i) \rho(l_i)}{\rho(\mathbf{x})} \\ \phi(g_i(\mathbf{x})) &= \mathcal{P}(\mathbf{x} | l_i) \rho(l_i) \\ \phi'(g_i(\mathbf{x})) &= \log \mathcal{P}(\mathbf{x} | l_i) + \log \rho(l_i) \end{aligned} \quad (2.24)$$

Isto simplifica bastante os cálculos quando assumimos que  $\mathcal{P}(\mathbf{x} | l_i)$  é uma distribuição com fator exponencial como, por exemplo, normal, Rayleigh, Poisson, etc. Considerando  $\mathcal{P}(\mathbf{x} | l_i)$  uma distribuição normal:

$$\mathcal{P}(\mathbf{x} | l_i) = \frac{1}{(2\pi)^{n/2} |\Sigma_i|^{1/2}} \exp \left[ -\frac{1}{2} (\mathbf{x} - \mu_i)^t \Sigma_i^{-1} (\mathbf{x} - \mu_i) \right] \quad (2.25)$$

onde  $\mathbf{x}$  e o valor esperado  $\mu_i$  na classe  $l_i$  são vetores de dimensão  $n$ ,  $\Sigma_i$  é a matriz de co-variância  $n \times n$  em  $l_i$ . Os parâmetros  $\mu$  e  $\Sigma$  da distribuição normal são os valores esperados de  $E[x]$  e  $E[(\mathbf{x} -$

$\mu_i)(\mathbf{x} - \mu_i)^t]$  para a média e co-variância de  $\mathbf{x}$  na classe  $l_i$ . Os pontos  $\mathbf{x}$  em  $l_i$  devem ser distribuídos em um hiperelipsóide em torno de  $\mu_i$ . A forma quadrática  $(\mathbf{x} - \mu_i)^t \Sigma_i^{-1} (\mathbf{x} - \mu_i)$  é a *distância de Mahalanobis* de  $\mathbf{x}$  a  $\mu_i$  (Equação 2.4). No caso unidimensional, em que  $\sigma_i^2 = |\Sigma_i|$  é a variância de  $\mathbf{x}$ , 99.7% das amostras estariam no intervalo  $|\mathbf{x} - \mu_i| \leq 3\sigma_i$ .

## 2.6 Outras abordagens

Outras abordagens para o problema de agrupamento são derivadas das abordagens apresentadas neste Capítulo. Novas formulações tentam eliminar pontos fracos e manter os pontos fortes de técnicas já desenvolvidas, e para tanto, muitas vezes certos aspectos da abordagem original é modificada ou combinada com outra já existente. Desde modo é possível perceber analogias entre técnicas aplicadas em contextos diferentes, sejam elas particularizando ou generalizando o procedimento de agrupamento.

Recentemente a técnica de *Mean-Shift* foi revisitada por Comaniciu e Meer [10] para aplicações em segmentação de imagem e vídeo. O Capítulo 4 apresenta uma nova abordagem para a técnica de *Mean-Shift* baseada em grafo. Para tanto será apresentado no Capítulo seguinte a Transformada Imagem-Floresta - IFT, que serve de base para a formulação teórica de uma nova técnica de agrupamento por Florestas Caminhos Ótimos - OPF. Esta técnica é uma extensão do princípio teórico utilizado na IFT, restrito ao domínio da imagem, para o espaço multidimensional de características.

## Capítulo 3

# Transformada Imagem-Floresta

A generalização do algoritmo da Transformada Imagem-Floresta (*Image Foresting Transform* — IFT) é descrita aqui juntamente com suas notações, definições, e aplicações correlacionadas como transformada *watershed* e busca de máximos regionais. Essencialmente a IFT é um extensão do algoritmo de caminhos mínimos de Dijkstra [45, 46, 47] para múltiplas fontes e funções gerais de caminhos. Em muitas aplicações práticas os custos de caminhos são inteiros com incrementos limitados e o grafo é esparso; neste caso, as otimizações descritas por Dial [48] e Ahuja et al. [49] se aplicam e resultam em uma execução em tempo linear no número de nós do grafo.

A IFT unifica e estende muitas técnicas de análise de imagens, mesmo sendo estas baseadas em conceitos similares que geralmente são apresentadas como métodos desconectados, como propagação ordenada, crescimento de regiões, inundação de bacias, dilatações geodésicas, programação dinâmica, busca em grafo  $A^*$ , etc. Todas essas técnicas podem transformar partições de um grafo em *zonas de influências* associadas com um conjunto semente, onde cada zona consiste de nós que são mais *fortemente conexos* à uma semente do que a qualquer outra. Estas zonas de influência são árvores de uma floresta definida pela IFT. Técnicas como a transformada *watershed* [50, 2] e segmentação *fuzzy-connected* [51, 52, 53, 54] são implementadas facilmente pela IFT.

### 3.1 Imagens como Grafos

A IFT utiliza o modelo de grafos baseando-se em duas propriedades importantes da imagem: uma *relação de adjacência* e uma *função de conectividade* entre pixels. O grafo é formado por pixels e seus arcos são formados pela relação de adjacência, o que define o conceito de conectividade entre pixels. A vizinhança entre pixels é definida de forma relativa à sua posição no conjunto de coordenadas dada pela relação de adjacência. O arco criado é ponderado por uma intensidade de conexão definida pela função de conectividade. Em geral, a função de conectividade visa extrair a informação similaridade

entre pixels. No entanto, cada operador de imagem pode utilizar esta relação entre pixels de forma diferente, sendo necessária a escolha da função apropriada. Desta forma, a IFT é um algoritmo que busca minimizar ou maximizar a função de conexão.

## 3.2 Notações e Definições

Formalmente, uma imagem  $\hat{I}$  pode ser definida como um par  $(\mathcal{N}, \vec{I})$  onde  $\mathcal{N} \subset Z^n$  é o domínio da imagem e  $\vec{I}(s)$  atribui um conjunto de  $c$  escalares  $I_i(s)$ ,  $i = 1, 2, \dots, c$ , para cada pixel  $s \in \mathcal{N}$ . Esta definição é genérica o bastante para imagens uma ou mais dimensões. Em imagens médicas provenientes de ressonância magnética, é comum trabalhar com imagens tridimensionais ( $\mathcal{N} \subset Z^3$ ), onde a terceira dimensão é construída pela sequência de imagens 2D (fatias).

Em geral, também é possível que mais de um escalar esteja associado a cada pixel da imagem, como por exemplo, em imagens coloridas onde  $c = 3$ . Características extraídas da imagem formam um vetor  $\vec{v}(s)$ , e no caso particular de bandas da imagem  $\vec{v}(s) = I(s)$ . Uma relação de adjacência  $\mathcal{A} \subset \mathcal{N} \times \mathcal{N}$  é utilizada para criar os arcos do grafo  $(\mathcal{N}, \mathcal{A})$ , onde o arco  $(s, t) \in \mathcal{A}$  indica que o pixel  $t$  é adjacente ao pixel  $s$ . Esta relação de adjacência geralmente é invariante à translação (Equações 3.1-3.3), mas ao levar em conta propriedades de imagem em  $\vec{v}$ , torna-se variante à translação (Equação 3.4).

- Adjacência Euclidiana:

$$(s, t) \in \mathcal{A} \quad \text{se} \quad \|t - s\| \leq h_i, \quad (3.1)$$

onde  $s = (x_s, y_s)$ ,  $t = (x_t, y_t)$ ,  $\|t - s\| = \sqrt{(x_t - x_s)^2 + (y_t - y_s)^2}$  e  $h_i > 0$  é o raio de um disco com centro em  $s$ .

- Adjacência retangular:

$$(s, t) \in \mathcal{A} \quad \text{se} \quad |x_t - x_s| \leq \frac{a}{2} \text{ e } |y_t - y_s| \leq \frac{b}{2}, \quad (3.2)$$

onde  $a > 0$  e  $b > 0$  são os lados de um retângulo com centro em  $s$ .

- Adjacência baseada em conjunto:

$$(s, t) \in \mathcal{A} \quad \text{se} \quad t - s \in \{(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)\}, \quad (3.3)$$

por exemplo:  $t - s \in \{(-2, 0), (-1, 0), (-1, -1), (0, -2), (1, 0), (2, 0)\}$ .

- Adjacência Euclidiana no espaço de características com restrição de domínio:

$$(s, t) \in \mathcal{A} \quad \text{se} \quad \|t - s\| \leq h_i \text{ e } \|\vec{v}(t) - \vec{v}(s)\| \leq h_f, \quad (3.4)$$

onde a adjacência definida por um raio  $h_f > 0$  no espaço de características e um raio  $h_i$  no domínio da imagem.

Os casos particulares de vizinhança-4 e vizinhança-8 são obtidos pela relação Euclidiana (Equação 3.1), com os valores de  $h_i = 1$  e  $h_i = \sqrt{2}$ , respectivamente. As relações definidas pelas Equações 3.3-3.4 podem ser assimétricas em uma ou mais dimensões. As relações de adjacência são ditas simétricas quando resultam em relação conexas simétricas entre nós no grafo. Em um grafo simétrico, todos os nós  $t$  pertencentes à adjacência  $\mathcal{A}(s)$  do nó  $s$  possuem uma adjacência simétrica  $\mathcal{A}(t)$  que contém este, conectando os nós  $s$  e  $t$  com arcos em ambas direções.

Um caminho  $\pi_s = \langle s_1, s_2, \dots, s_n \rangle$  em  $(\mathcal{N}, \mathcal{A})$  é uma sequência de nós adjacentes e duas amostras são *conexas* se existe um caminho entre eles. O valor de função de caminho  $f(\pi_s)$  é atribuída a qualquer caminho  $\pi$  em um grafo. O caminho  $\pi_s$  é *trivial* se  $\pi_s = \langle s_1 \rangle$  e denota-se  $\pi_s \cdot \langle s, t \rangle$  a concatenação do caminho  $\pi_s$  com sufixo em  $s$  e o arco  $(s, t)$ , como ilustra o exemplo da Figura 3.1a. Os caminhos são representados por arcos direcionados em sentido contrário ao caminho, como ilustra a Figura 3.1d, onde  $P(t)$  é nó predecessor de  $t$  no caminho  $\pi_t$  e  $R(t)$  é o nó inicial. Um mapa de predecessores  $P$  é uma função que atribui a cada amostra  $s \notin R$  seu predecessor  $P(s)$  em um caminho ótimo de  $R$  ou um marcador *nulo* quando  $s \in R$ .

### 3.2.1 Caminhos Ótimos

A função de conectividade atribui um valor  $f(\pi_t)$  para qualquer caminho  $\pi_t$ , baseado em propriedades da imagem. Um caminho  $\pi_t$  é *ótimo* se  $f(\pi_t) \leq f(\tau_t)$  (ou  $f(\pi_t) \geq f(\tau_t)$ ) para qualquer outro caminho  $\tau_t$ , independentemente de sua raiz. O valor de caminho pode ser minimizado ou maximizado pela atribuição de caminhos ótimo  $\pi_t$  para cada pixel  $t \in \mathcal{N}$ .

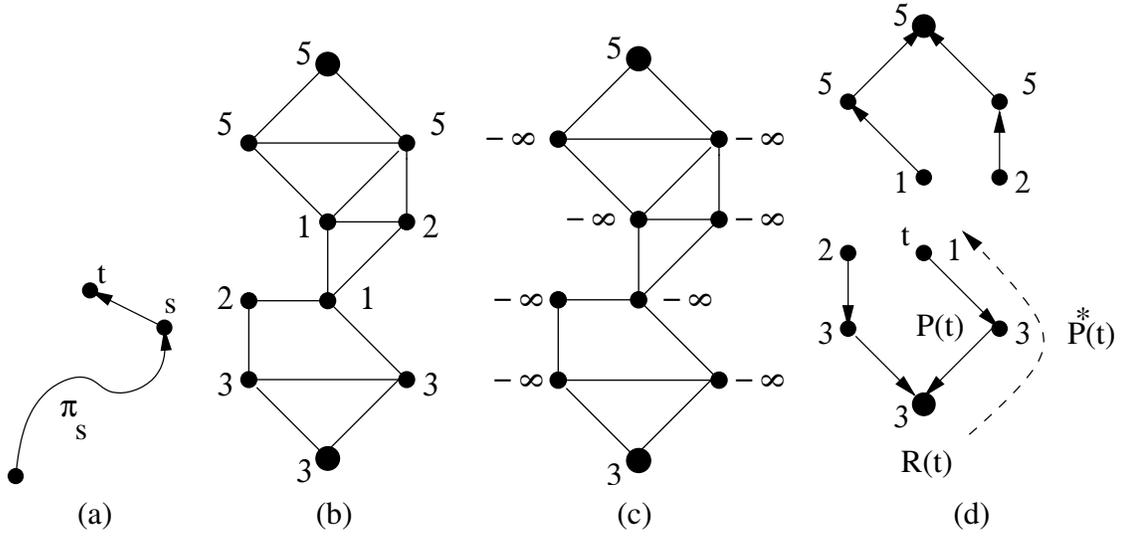


Fig. 3.1: Um caminho  $\pi_s$  estendido por um arco  $\langle s, t \rangle$ . (b) Um grafo com pesos no nós, onde existem dois máximos com valores 3 e 5. Os nós maiores indicam o conjunto semente  $S$ . (c) Caminhos triviais  $f_{min}(\langle t \rangle)$  para cada nó  $t$ . (d) Floresta de caminhos ótimos  $P$  para  $f_{min}$  e os valores de fim de caminho  $V(t)$ . O caminho ótimo  $P^*(t)$  dado pela Equação 3.8 (linha pontilhada) pode ser obtido percorrendo os predecessores até a raiz  $R(t)$  para cada amostra.

$$V(t) = \min_{\forall \pi_t \in (\mathcal{I}, \mathcal{A})} \{f(\pi_t)\} \quad (3.5)$$

$$V(t) = \max_{\forall \pi_t \in (\mathcal{I}, \mathcal{A})} \{f(\pi_t)\} \quad (3.6)$$

onde  $V(t)$  é o valor de caminho ótimo com término em  $t$ .

Uma *floresta espalhada* é um mapa de predecessores  $P$  que não contém ciclos e todos seus caminhos são ótimos. Para cada pixel  $t \in \mathcal{N}$ , uma floresta espalhada  $P$  define um caminho ótimo  $P^*(t)$  que pode ser facilmente obtido seguindo o caminho de  $P(t)$  até sua raiz  $R(t)$  em  $S$ . A floresta de caminhos ótimos é definida pelo mapa de predecessores  $P$  com raízes em um conjunto  $R \subset \mathcal{N}$ . A IFT soluciona o problema de otimização pelo mapeamento de um imagem  $\hat{I}$  no grafo  $(\mathcal{N}, \mathcal{A})$  e este em uma floresta de caminhos ótimos  $P$  de acordo com a função de caminho dada.

Qualquer função de caminho pode ser utilizada pelo algoritmo da IFT desde que seja *suave*, isto é, para qualquer amostra  $t$ , existe um caminho ótimo  $\pi_t$  o qual é trivial ou possui a forma  $\tau_s \cdot \langle s, t \rangle$ , onde

- $f(\tau_s) \leq f(\pi_t)$ ;
- $\tau_s$  é ótimo, e

- para qualquer caminho ótimo  $\tau'_s$ ,  $f(\tau'_s \cdot \langle s, t \rangle) = f(\pi_t)$ .

As funções de conectividade são especificadas por uma regra de inicialização e por uma regra de extensão de caminho. Vários exemplos de funções de custo (Equações 3.7-3.10) podem ser definidos através de diferentes funções de caminho e um valor inicial de caminho trivial em  $s$ , dado por um  $H(s)$ . Em aplicações típicas da IFT, normalmente restringe-se a busca por caminhos que se originam em um conjunto  $S \subset \mathcal{N}$  de nós, denominados de *sementes*. A definição do conjunto  $S$  pode ser dada como entrada do algoritmo ou definida na função de caminho.

$$\begin{aligned} f_{\max}(\langle t \rangle) &= H(s) \\ f_{\max}(\pi_s \cdot \langle s, t \rangle) &= \max\{f_{\max}(\pi_s), w(s, t)\} \end{aligned} \quad (3.7)$$

$$\begin{aligned} f_{\min}(\langle t \rangle) &= H(s) \\ f_{\min}(\pi_s \cdot \langle s, t \rangle) &= \min\{f_{\min}(\pi_s), w(s, t)\} \end{aligned} \quad (3.8)$$

$$\begin{aligned} f_{\text{sum}}(\langle t \rangle) &= H(s) \\ f_{\text{sum}}(\pi_s \cdot \langle s, t \rangle) &= f_{\text{sum}}(\pi_s) + w(s, t) \end{aligned} \quad (3.9)$$

$$\begin{aligned} f_{\text{euc}}(\langle t \rangle) &= \begin{cases} 0 & \text{se } t \in \mathcal{S} \\ +\infty & \text{caso contrário} \end{cases} \\ f_{\text{euc}}(\pi_s \cdot \langle s, t \rangle) &= \|t - R(s)\|^2 \end{aligned} \quad (3.10)$$

onde  $H(s)$  é o valor inicial,  $w(s, t)$  é um peso do arco  $\langle s, t \rangle$ , e  $\mathcal{S} \subset \mathcal{N}$  é o conjunto semente. Funções  $f_{\max}$ ,  $f_{\text{sum}}$ , e  $f_{\text{euc}}$  devem ser minimizadas (Equação 3.5) enquanto a função  $f_{\min}$  deve ser maximizada (Equação 3.6). Funções  $f_{\max}$  e  $f_{\min}$  são usadas para operações duais, e  $w(s, t) \geq 0$  em  $f_{\text{sum}}$ . A restrição ao conjunto semente em  $f_{\text{euc}}$  também pode ser aplicada a  $f_{\min}$ ,  $f_{\max}$ , e  $f_{\text{sum}}$  pela atribuição de valores finitos de  $H(s)$  a pixels  $s \in \mathcal{S}$ .

Cada semente define uma *zona de influência* composta por pixels conexos a ela por caminhos com menor custo do que quaisquer outros fornecidos por qualquer outra semente em  $S$ . A Figura 3.1b mostra um exemplo de um grafo com nós ponderados, onde os valores do peso estão ao lado e nós maiores pertencem ao conjunto de semente. Os valores iniciais de  $H$  são atribuídos aos nós sementes na Figura 3.1b e  $+\infty$  aos nós restantes. O caminho ótimo  $P^*$  dado pela função  $f_{\max}$  da Equação 3.6 a partir do nó  $t$  no grafo até sua raiz  $R(t)$  é ilustrado através da seta pontilhada na Figura 3.1d.

### 3.3 Aplicações da IFT

A IFT unifica diversas técnicas relacionadas com processamento de imagens cujos conceitos são similares, ou seja, são baseados em propagação, dilatações geodésicas, crescimento de regiões, procura em grafo, programação dinâmica, etc. Geralmente estas técnicas são apresentadas como métodos desconectados, mas sua unificação dentro de um mesmo arcabouço provê um melhor entendimento de suas relações entre si [55, 2, 56, 57], e permite extensão de novos operadores. A IFT também permite uma implementação em *hardware* [58], diferencial [59] e paralela [60] ainda que sua complexidade seja linear em grande parte de seus operadores de imagens. A formulação de operadores de imagem a partir da IFT requer pouca alteração no seu algoritmo geral e em alguns casos resulta em uma implementação mais robusta e flexível do método. A IFT unifica reconstruções morfológicas [21, 61] e transformadas *watershed* [62, 50, 63] em uma única transformação [55, 19]. No contexto deste trabalho, a reconstrução morfológica e a transformada *watershed* são combinados e estendidos para o domínio multidimensional de características para prover uma solução robusta e flexível de agrupamento por floresta de caminhos ótimos (Seção 4.1).

O algoritmo geral da IFT (Algoritmo 1) utiliza como princípio a competição entre sementes para criar uma floresta de caminhos ótimos. Os mapas de raízes  $R$  pode ser calculado a partir  $P$  durante ou após a execução do algoritmo. A implementação da fila de prioridades  $Q$  determina a complexidade do algoritmo. Uma implementação simples poderia tornar seu tempo de execução  $O(|\mathcal{I}|^2)$ , mas pode ser reduzido para  $O(|\mathcal{A}| + |\mathcal{I}| \log |\mathcal{I}|)$  se usada uma estrutura de dados *heap* binária [64]. A variante do algoritmo de Dijkstra proposta por Dial [48] para caso especial em que os pesos dos arcos são inteiros no intervalo  $[0..K]$  permite sua implementação através de *bucket sorting* e reduz seu tempo de execução para a complexidade linear  $O(|\mathcal{A}| + |\mathcal{I}|K)$ . A solução de Dial consiste em um vetor circular de ponteiros para a lista duplamente ligada de pixels (Figura 3.2a), o que facilita sua manipulação. As operações de inserção e atualização podem ser feitas em  $O(1)$  enquanto a remoção de pixels é feita em  $O(K)$  devido a procura no vetor. O custo de alocação dinâmica pode ser reduzido se for utilizado o vetor auxiliar pré-alocado com tamanho  $|\mathcal{I}|$  para armazenar as listas [1] (Figura 3.2b). No caso de empate de custos, a fila obedece uma política de desempate baseada em FIFO - *First-In First-Out* ou LIFO - *Last-In First-Out*, que pode ser escolhida de acordo com a operação desejada. A política FIFO é mais utilizada em grande parte das aplicações, enquanto a política LIFO pode ser utilizada para criar árvores espalhadas mínimas.

**Algoritmo 1** – IFT

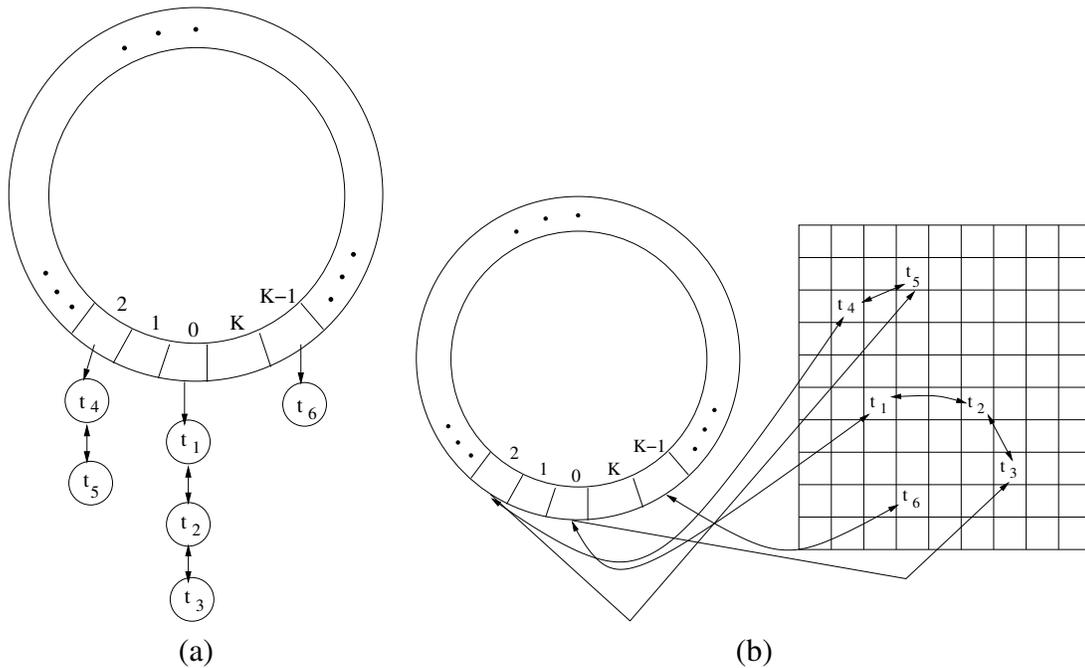


Fig. 3.2: (a) Estrutura de Dial para a fila  $Q$ . (b) A estrutura de dados proposta em [1].

ENTRADA: Uma imagem, uma relação de adjacência  $A$ , um conjunto de nós sementes  $S$  e uma função suave de valor de caminho  $f$ .

SAÍDA: Mapa de valores de custo de caminhos  $V$ , mapa de predecessores  $P$  e mapa de raízes  $R$ .

AUXILIARES: Fila de prioridades  $Q$  inicialmente vazia e variável  $cst$ .

1. **Para Cada** nó  $s$  do grafo, **Faça**
2.      $P(s) \leftarrow nil, R(s) \leftarrow s$  e  $V(s) \leftarrow f(\langle p \rangle)$ .
3.     **Se**  $V(s)$  for finito, **Então**
4.         Insira  $s$  em  $Q$ .
5. **Enquanto**  $Q$  não for vazia, **Faça**
6.     Remova  $s$  de  $Q$  tal que  $V(s)$  é mínimo.
7.     **Para Cada** nó  $t \in A(s)$  tal que  $V(t) > V(s)$ , **Faça**
8.          $cst \leftarrow f(P^*(s) \cdot \langle s, t \rangle)$ .
9.         **Se**  $cst < V(t)$ , **Então**
10.              $P(t) \leftarrow s, R(t) \leftarrow R(s)$  e  $V(t) \leftarrow cst$ .
11.             Atualize  $t$  em  $Q$ .

As linhas 1 – 4 inicializam a floresta como um conjunto de árvores triviais, nós isolados a serem conquistados durante o processo. Os custos são iniciados com  $f(\langle p \rangle)$  refletindo que nenhum caminho

a partir das sementes foi processado. No caso da função  $f_{max}$  (Equação 3.7), por exemplo,  $V(s) \leftarrow 0$ , se  $s \in S$ , e  $V(s) \leftarrow +\infty$ , caso contrário. Assim, os caminhos triviais a partir das sementes são inicializados. Tais caminhos possuem o custo mínimo 0 (para o caso da função  $f_{max}$ , por exemplo) de forma que todas as sementes se tornarão, obrigatoriamente, raízes da floresta. As sementes são inseridas na fila de prioridades  $Q$  (linhas 3 – 4). Os pixels presentes na fila de prioridades representam a fronteira da floresta em crescimento, os quais correspondem a nós da floresta atingidos por caminhos não necessariamente ótimos. A cada iteração do algoritmo (linha 5) um caminho ótimo é selecionado, o qual corresponde ao caminho de menor custo entre os nós que atingem a fronteira da floresta (linha 6). No caso de empate dos custos do nós, a fila  $Q$  utiliza um critério de desempate. Os vértices adjacentes à fronteira do caminho são avaliados (linha 7). A fronteira pode ser ampliada pela aquisição de novas conexões ou melhores rotas podem ser encontradas para pixels de fronteira já existentes. Na linha 8 é calculado o custo  $cost$  de uma nova possível rota, o qual é comparado com o custo do caminho atual (linha 9). Os mapas  $V$  e  $P$  devem ser atualizados de forma a refletirem o melhor caminho encontrado (linha 11). A condição  $V(t) > V(s)$  na linha 7 é uma otimização que explora o fato de o custo ao longo do caminho ótimo pode não ser decrescente. Assim sendo, quando existem várias sementes em  $S$ , estas serão propagadas pela política FIFO e teremos um processo competitivo.

Figura 3.3a ilustra um exemplo simples de um grafo 4-conexo, com um objeto sombreado no centro. Uma função de dissimilaridade atribui pesos mais altos para arcos que cruzam a borda do objeto do que arcos dentro e fora deste. Seja  $(0, 0)$  a coordenada  $(x, y)$  da esquina superior esquerda, valores iniciais de custo são dados por  $f_{max}$  (Equação 3.7), onde  $H(t) = 0$ , se  $t \in S = \{(2, 2), (4, 0)\}$ , e  $H(t) = +\infty$  caso contrário. O valor de caminho é o valor máximo de arco ao longo do caminho. A semente em  $(2, 2)$  para pixels internos oferece caminhos com menor valor do que a semente externa em  $(4, 0)$ , e o contrário ocorre para pixels externos. A minimização de  $f_{max}$  pelo Algoritmo 1 separa o objeto e o fundo em duas árvores, cada uma enraizada em uma semente (Figure 3.3i). Esse processo dura 25 iterações do laço principal do Algoritmo 1. Nas duas primeiras iterações, as sementes são removidas de  $Q$ , elas conquistam seus vizinhos e os inserem em  $Q$  (Figuras 3.3b-c). A semente interna propaga caminhos para todos os pixels internos e alguns pixels no fundo nas iterações iniciais (Figuras 3.3d-f). Após 12 iterações, a semente externa começa a reconquistar os pixels do fundo até que a floresta final é formada (Figuras 3.3g-i).

### 3.3.1 Operadores na Superfície da Imagem

Uma imagem  $\hat{I} = (\mathcal{N}, I)$  pode ser considerada como uma superfície discreta, onde os níveis da superfície correspondem ao valor de intensidade  $I$ , constituída de domos, bacias e platôs. Domos e bacias são formados pela zona de influência de máximos e mínimos locais, onde o valor de intensidade

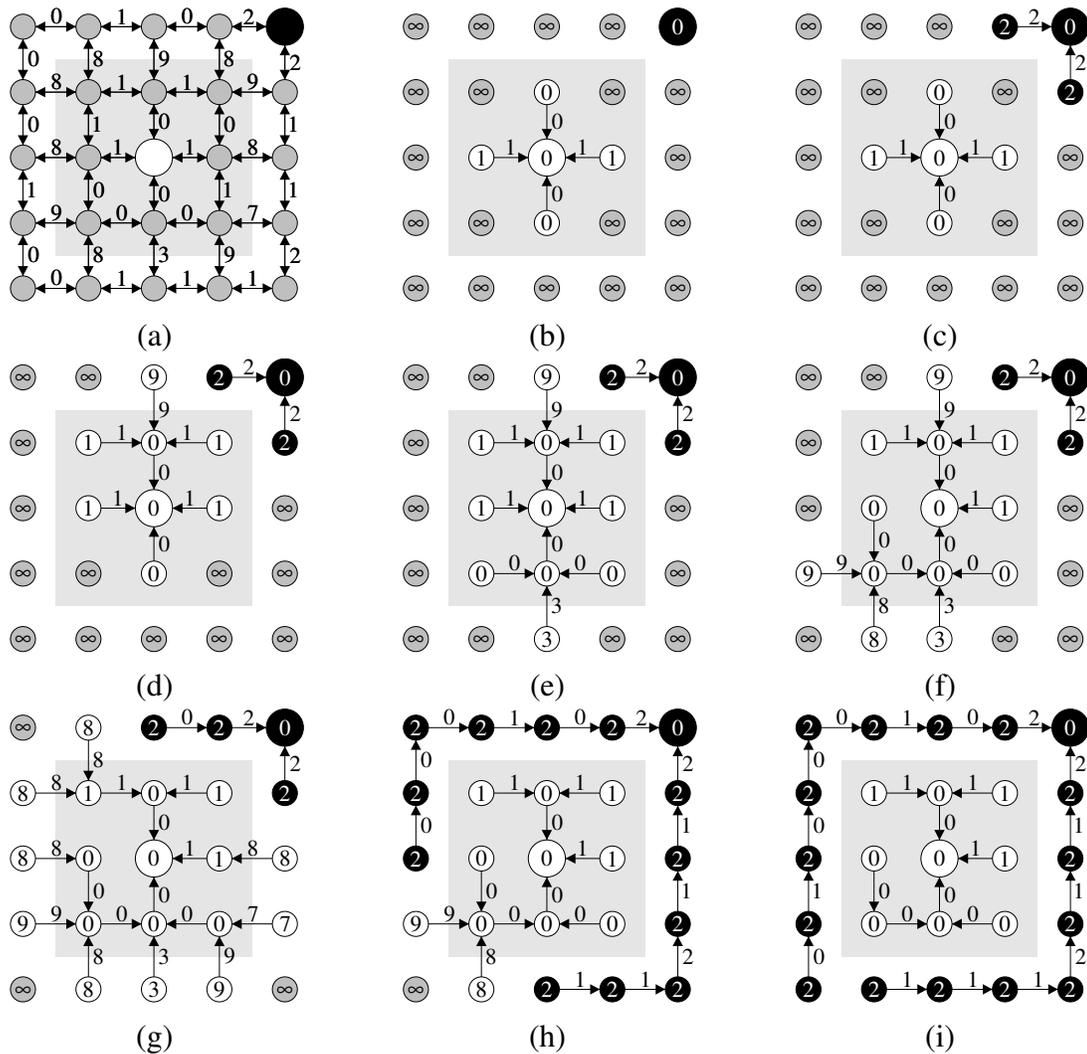


Fig. 3.3: (a) Um grafo 4-conexo com uma semente (2, 2) dentro de um objeto sombreado e uma semente (4, 0) no fundo. (b)-(f) Floresta após 5 iterações, onde a semente interna está conquistando e inserindo pixels de fundo em  $Q$ . (g) Floresta após 12 iterações. A semente externa começa a reconquistar os pixels de fundo. (h) Floresta após 20 iterações. (i) Floresta final após 25 iterações.

da região conexa é estritamente maior e menor que seus pixels adjacentes, respectivamente. Ou seja, um *máximo regional* é o conjunto conexo máximo  $N' \subseteq N$ , tal que  $I(s) \geq I(t)$  para qualquer  $s \in N$  e qualquer arco  $(s, t) \in A$  [21]. Um *mínimo regional* é definido de maneira análoga, onde  $I(s) \leq I(t)$ , e um *platô* é formado por um conjunto conexo onde  $I(s) = I(t)$ . Um platô pode fazer parte de uma bacia ou domo, ou ainda ser um máximo ou mínimo regional.

### 3.3.2 Fechamento de Bacias e Domos

Reconstruções morfológicas são utilizadas para simplificar ou remover ruídos de imagens, e suas operações básicas são dilatação e erosão. Estas quando combinadas nessa ordem provem uma filtragem por fechamento e na ordem inversa provem filtragem por abertura. Estas operações são mais comuns em imagens binárias, onde buracos são fechados por fechamento e componentes conexos são removidos por abertura, de acordo com o tamanho da adjacência utilizada. Em imagens nível de cinza, estas operações preenchem bacias ou removem topos de domos. A operação morfológica é realizada durante a execução do algoritmo de *watershed* de marcadores de rotulados, através do mapeamento do custos de caminhos no brilho da imagem final.

Uma imagem dada por  $\hat{I} = (\mathcal{N}, I)$  e uma relação de adjacência simétrica  $\mathcal{A}$ , define o grafo  $(\mathcal{I}, \mathcal{A})$ , como visto na Seção 3.1. Seja  $\mathcal{S} \in \mathcal{I}$  um subconjunto de pixels um um máximo  $\hat{I}$ , a partir dos quais caminhos ótimos  $\pi_s$  são criados até todo pixel  $s \in \mathcal{I}$ . Para cada caminho, o máximo entre seu valor e a intensidade do pixel é mínimo ao longo deste (Equação 3.8). Os caminhos competem entre si para minimizar  $V(s)$  através da função de caminho  $f_{bacias}$ , que é um caso particular de  $f_{max}$ .

$$\begin{aligned} f_{bacias}(\langle s \rangle) &= H(s) \\ f_{bacias}(\pi_s \cdot \langle s, t \rangle) &= \max\{f_{\max}(\pi_s), I(t)\}, \end{aligned} \quad (3.11)$$

onde

$$H(s) = \begin{cases} I(s) & \text{se } s \in \mathcal{S} \\ +\infty & \text{caso contrário.} \end{cases} \quad (3.12)$$

O resultado de  $(\mathcal{I}, V)$  é uma filtragem da imagem onde bacias de  $\hat{I}$  são preservadas e aquelas marcadas por  $\mathcal{S}$  são preenchidas com altura  $H(s)$ . No caso análogo, a função de caminho  $f_{domos}$  é dada pelo caso particular de  $f_{min}$ .

A Figura 3.4 ilustra um exemplo de reconstrução por marcadores rotulados, onde a intensidade da imagem é representada por barras brancas sobrepostas ao valor de caminho  $H(s)$  representado por barras cinzas. A inicialização do processo de reconstrução através de marcadores binários utiliza  $H(s)$  com valor inicial zero no conjunto semente  $\mathcal{S}$  e  $+\infty$  caso contrário (Figura 3.4a). O resultado é a floresta de caminhos ótimos (Figura 3.4b), onde rótulos diferentes (sólido e rachurado) são atribuídos a cada árvore de  $P(s)$ , bacias são preenchidas por  $V(s)$  e o domo entre bacias é dividido igualmente pela política FIFO. Em outro exemplo ilustrado na Figura 3.4c, sementes são utilizadas em todos os pixels com valores nível de cinza dados por  $H(s) > I(s)$ . A função  $H(s)$  tem valores finitos somente onde existem sementes rotuladas em  $\mathcal{S}$ . Neste caso, apenas uma árvore conquista todos os pixels na

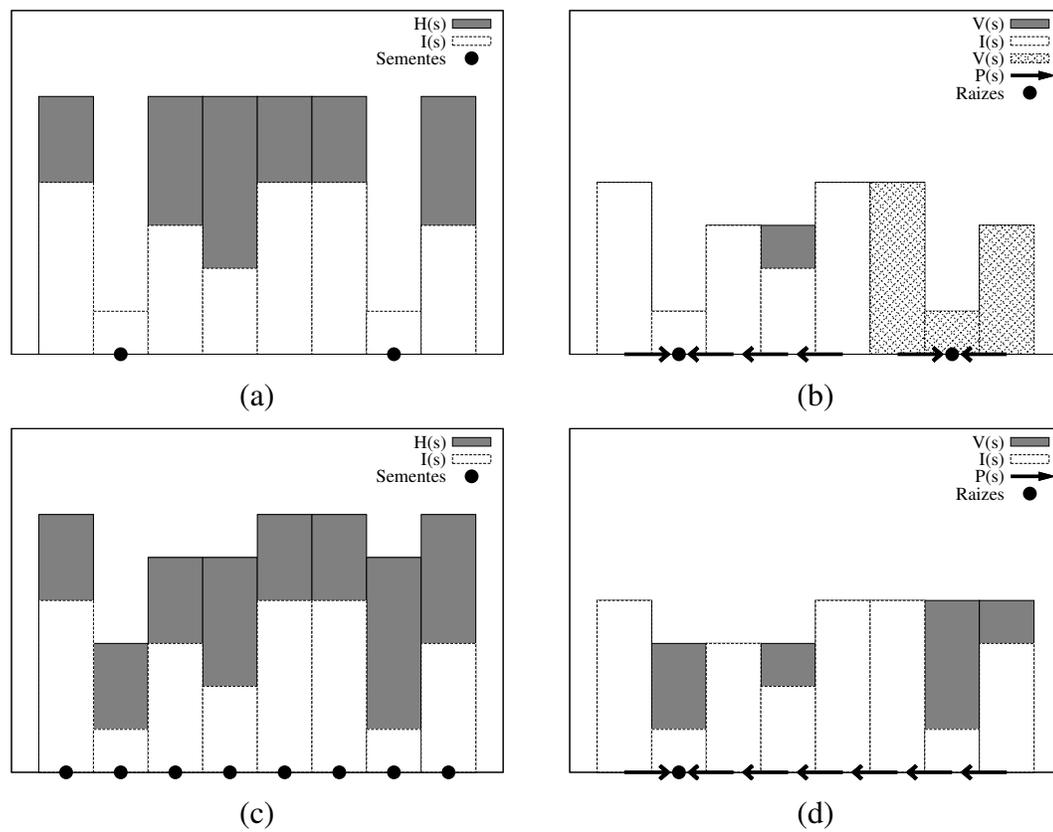


Fig. 3.4: Um espaço unidimensional onde a intensidade de  $I(s)$  é representada por barras brancas sobrepostas sobre barras cinzas de  $H(s)$ . (a) Sementes são utilizadas como marcadores binários, onde o custo inicial  $H(s) = I(s)$  nas sementes e  $H(s) = +\infty$ , caso contrário. (b) O resultado da IFT em (a) preenche bacias com valor de  $V(s)$  e duas árvores são formadas com rótulos sólido e rachurado. (c) Sementes são utilizadas em todos os pixels com valores  $H(s) > I(s)$ . (d) Caminhos criados em (c) competem entre si para criar uma única árvore que conquista todos os pixels e preenche três bacias.

Figura 3.4d, preenchendo as bacias com o valor mínimo de caminho  $V(s)$ . A função  $V(s)$  também é chamada de reconstrução morfológica superior de  $I(s)$  a partir da função marcadora  $H(s)$ .

### 3.3.3 A Transformada *Watershed*

O algoritmo de *watershed* utiliza o fechamento de bacias para simular um preenchimento na superfície do gradiente da imagem por águas que rotulam objetos de acordo com o conjunto semente. Linhas divisórias são criadas entre águas de rótulos diferentes. Os objetos de interesse no grafo são delimitados pelas cristas que são mais altas que qualquer domo dentro ou fora dos objetos. Neste caso, os objetos podem ser separados através da escolha de nós dentro de cada objeto e da partição que constitui o fundo da imagem. Marcadores são rotulados como nós sementes internos e externos,

formando um conjunto  $S$ , onde cada objeto recebe um rótulo distinto. O algoritmo da IFT é executado no grafo da imagem  $(\mathcal{N}, \mathcal{A})$  de modo a minimizar a função de conectividade  $f_{max}$ , definida na Equação 3.7, na superfície dada por  $I(s)$ . A conectividade entre pixels é restrita pela adjacência  $A_8$  que conecta os pixels vizinhos-8 através de  $h_i = \sqrt{2}$  na Equação 3.1. Este conceito da adjacência é o mesmo de um elemento estruturante plano em morfologia matemática. O mapa de rótulos  $L$  dá o resultado da segmentação.

Essa formulação captura os aspectos essenciais da transformada *watershed* (WT — *Watershed Transform*) e suas variações [65, 62, 50, 20]. Não existia uma definição consensual para a transformada de WT [63] até a formulação apresentada por Romaric et al [66]. A descrição é usualmente feita de forma informal, como o processo de inundação da superfície da imagem, com uma fonte de água em cada semente; uma barreira divisora de águas é construída onde duas águas com rótulos diferentes se encontram. No entanto, a posição da divisões de água não são precisamente definidas em muitas situações, como platôs e quando duas massas de água inundam uma terceira bacia ao mesmo tempo. Com a abordagem de IFT, essa ambiguidade é decidida pelo critério de desempate. O critério da FIFO geralmente leva a uma partição igualmente distribuída entre platôs e bacias através das competição das fontes.

O Algoritmo 2 descreve o procedimento de fechamento de bacias e watershed, executados simultaneamente. A imagem de entrada  $I$  é a imagem de brilho ou a imagem de gradiente, se o objetivo é filtragem por reconstrução ou rotulamento de objetos por *watershed*.

#### Algoritmo 2 – FECHAMENTO DE BACIAS E WATERSHED

ENTRADA: Imagem  $\hat{I} = (\mathcal{I}, I)$ , adjacência  $\mathcal{A}$ , função de rotulamento  $\lambda(t)$  e função de custo inicial  $H(t)$ .  
 SAÍDA: Função de imagem filtrada  $V$ , mapa de rótulos  $L$  e floresta de caminhos ótimos  $P$ .  
 AUXILIARES: Fila de prioridade  $Q$  e variável  $tmp$ .

1. **Para Cada**  $t \in \mathcal{I}$ , **Faça**  $P(t) \leftarrow nil$  e  $V(t) \leftarrow H(t)$ . **Se**  $V(t) \neq +\infty$ , **Então** insira  $t$  em  $Q$ .
2. **Enquanto**  $Q \neq \emptyset$ , **Faça**
3.     Remova  $s$  de  $Q$  de forma que  $V(s)$  é mínimo.
4.     **Se**  $P(s) = nil$ , **Então**  $V(s) \leftarrow I(s)$  e  $L(s) \leftarrow \lambda(s)$ .
5.     **Para Cada**  $t \in \mathcal{A}(s)$ , tal que  $V(t) > V(s)$ , **Faça**
6.         Calcule  $tmp \leftarrow \max\{V(s), I(t)\}$ .
7.         **Se**  $tmp < V(t)$ , **Então**
8.             **Se**  $V(t) \neq +\infty$ , **Então** remova  $t$  de  $Q$ .
9.             **Faça**  $P(t) \leftarrow s$ ,  $V(t) \leftarrow tmp$ ,  $L(t) \leftarrow L(s)$  e insira  $t$  em  $Q$ .

Modificando a função de caminho  $f_{max}$  para usar o conjunto semente  $S$  na inicialização de caminhos triviais, a IFT calcula a transformada *watershed* de marcadores (WMT — *Watershed-from-Markers Transform*). Se os custos iniciais utilizados forem dados por  $H(s) > I(s)$  para todos os nós  $s \in S$ , podemos calcular a WMT sem imposição de marcadores [62] — isto é, algumas sementes não serão raízes da floresta [20]. Esta solução é mais eficiente que a solução original baseada na mudança de homotopia [62].

Os custos iniciais dados por  $H(s)$  podem ser criados a partir de diversas operações para tornar  $H(s) > I(s)$ , como por exemplo  $H(s) = I(s) + 1$ . O número de mínimos pode ser reduzido ainda mais se o raio  $h_i$  da adjacência for aumentado. As Figura 3.5 ilustra o resultado da segmentação dado pro  $L(s)$ , onde o objetivo é delinear veias e ossos, para raios  $h_i = 2.5$  (Figura 3.5a) e  $h_i = 3.5$  (Figura 3.5b)

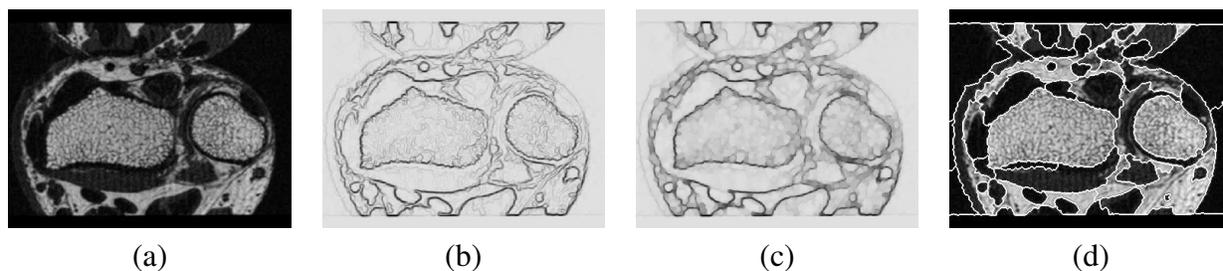


Fig. 3.5: (a) Imagem de ressonância magnética do pulso. (b) Uma função gradiente de  $I(t)$  e (c) seu fechamento  $I(s) + 1$  (invertidos para ilustração) usando uma adjacência de raio  $h_i = 2.5$ . (d) Função  $L(t)$  é calculada a partir de adjacência de raio  $h_i = 3.5$ .

## Capítulo 4

# Agrupamento por Floresta de Caminhos Ótimos

Abordagens clássicas de agrupamento por grafos (Seção 2.1.2) interpretam amostras como nós de grafo completo, cujo peso do arco são as distâncias entre amostras. Esses métodos constroem um sub-grafo de vizinhança, como uma *minimum spanning tree* ou grafo de Gabriel, e removem arcos inconsistentes baseado em um critério, de natureza algumas vezes hierárquica (Seção 2.1.1).

Em análise de imagens, o grafo é geralmente esparsos em que pixels são nós e arcos são definidos por uma relação de adjacência no domínio da imagem, como por exemplo vizinhança de conectividade-4. Os arcos são ponderados por valores de similaridade e o problema de agrupar pixels torna-se um problema de segmentação, cuja solução é obtida pela partição ótima do grafo. Diferentes medidas de cortes em grafo tem sido utilizados nesse contexto. O primeira medida foi a soma dos pesos dos arcos ao longo do corte [18], mas sua tendência de criar pequenos grupos encaminham outras medidas corte médio [28], corte na média [29], associação média [30], corte normalizado [31], corte proporcional [32] e corte por funções de energia [67, 35].

Outras abordagens para agrupamento de dados exploram a densidade de probabilidade (PDF), que pode se estimada por uma janela de Parzen (Seção 2.1.2). Algumas de abordagens assumem explicitamente ou, geralmente, implicitamente que a PDF de um formato conhecido e tenta estimar seus parâmetros [4, 6, 26]. No entanto, grupos podem ter formas um tanto diferentes da forma hiperelíptica, que é a suposição clássica. Vários métodos visam obter grupos com formas arbitrárias [68, 8, 9, 69]. Entre estes, o algoritmos de *mean-shift* tem sido o mais popular nos últimos anos [8, 12, 10, 11, 70, 14, 15, 71].

O algoritmo de floresta de caminhos ótimos (OPF) foi primeiramente utilizado por Papa et al. [72, 73] em classificação supervisionada de padrões. A técnica de Papa et al. utiliza uma MST do grafo completo e removendo-se os arcos entre classes diferentes, tais amostras adjacentes tornam-se pro-

tótipos. Esta abordagem tem sido amplamente utilizado em diversas aplicações, tais como avaliação de descritores de textura [74, 75], identificação de disfagias (dificuldade de deglutição) em seres humanos [76] e diagnóstico automático de patologias na laringe [77]. Em aplicações médicas na área de parasitologia, OPF foi utilizado em duas patentes, uma delas nacional [78] e a outra internacional [79], onde a finalidade é identificar parasitos intestinais em seres humanos.

A proposta de um arcabouço híbrido de agrupamento de dados, que combina particionamento de grafo por OPF visa suprir limitações de métodos de agrupamento baseado em modelos paramétricos e expandir o uso da IFT para o espaço de características. As amostras são nós de um grafo não-completo, cujos arcos são definidos por uma *relação de adjacência*. Essa relação considera os  $k$ -vizinhos mais próximos que estão a uma distância máxima entre amostras no espaço de características. No contexto de agrupamento de pixels, essa distância também considera restrição de conectividade de domínio da imagem. Os pesos dos arcos são as distâncias entre amostras para definir a similaridade entre nós. Estes são ponderados pelos valores de sua densidade de probabilidade, estimada pelos pesos das arestas. Um caminho é uma sequência de nós adjacentes e a *função de caminho* estima a força de conectividade entre seus nós terminais. Desta forma, é atribuído ao nó terminal  $s$  de cada caminho o mínimo entre os valores de densidade ao longo do caminho e o seu valor de custo inicial. Os custos iniciais funcionam como parâmetros de filtragem na PDF, reduzindo o número de grupos que possuem densidade superiores a um limiar. A maximização de valores de caminho para cada amostra  $s$ , independentemente de seu nó de início (raiz), particiona o grafo em uma *floresta de caminhos ótimos*. As raízes da floresta formam um subconjunto dos máximos da PDF, onde cada raiz define uma árvore de caminhos ótimos (grupo ou zona de influência do respectivo máximo) composto pelas amostras mais fortemente conectadas.

O método de não-supervisionado de agrupamento baseado em OPF estende a Transformada Imagem-Floresta (IFT) — uma ferramenta de desenvolvimento de operadores de processamento de imagem baseado em conectividade [19] — do domínio da imagem para o espaço de características. Desta forma revela relações entre o algoritmo de *mean-shift* [8], esqueleto por zona de influência no espaço de característica [9, 69], reconstruções [80, 21, 20], transformações *watershed* [81, 82, 50, 2] e segmentação *fuzzy-connected* [83, 52]. Também provê uma implementação mais geral e robusta do popular algoritmo de *mean-shift* [8]. Os resultados do método são ilustrados no contexto de segmentação de imagens.

## 4.1 Método Proposto

Nesta seção serão abordadas as alterações no algoritmo original da IFT para sua aplicação de agrupamento no espaço de características. O custo de um caminho é substituído por um valor de

caminho, que é proporcional à estimativa de densidade em um adjacência da amostra. O valor de caminho que deve ser maximizado em direção às modas através de uma função de caminho que encontra os máximos regionais da PDF.

### 4.1.1 Estimação de Densidade

Seja  $(N, A)$  o grafo definido na Seção 3.2. O mapeamento que pondera os nós dos grafos é dado pela estimativa de densidades  $\rho$ .

$$\rho(s) = \frac{1}{\sqrt{2\pi\sigma^2}|\mathcal{A}(s)|} \sum_{\forall t \in \mathcal{A}(s)} \exp\left(\frac{-d^2(s, t)}{2\sigma^2}\right) \quad (4.1)$$

Isto é equivalente a uma estimação de densidade com uma janela de Parzen usando o kernel Gaussiano[17]. Dado que aproximadamente 99.7% das amostras em uma função Gaussiana está abaixo de  $d(s, t) = 3\sigma$  e que a distância máxima entre duas amostras adjacentes é  $d_f$ , como mostra a Equação 4.2, podemos definir  $\sigma = d_f/3$  para garantir que todas as amostras adjacentes serão consideradas dentro do kernel.

$$d_f = \max_{\forall (s, t) \in \mathcal{A}} \{d(s, t)\} \quad (4.2)$$

A largura de banda  $d_f$  do kernel pode ser estimada como proposta em [70, 44, 43, 84]. No entanto, o método tradicional de estimar a PDF pela janela de Parzen pode apresentar problemas em diferentes escalas e concentrações de amostras. Soluções para o problema levam a escolhas adaptativas de  $d_f$  dependendo da região do espaço de características. Levando em consideração uma relação de adjacência  $A$  dada pelos  $k$ -vizinhos mais próximos (Equação 4.4) é possível considerar diferentes concentrações [85]. Isto reduz o problema de escala à busca de um melhor valor de  $k$  dentro de um intervalo de escala. Como resultado, temos um grafo de  $k$ -vizinhos mais próximos para um dado valor de  $k$ , cujos pesos  $\rho(s)$  pela Equação 4.1 são ilustrados pela Figura 4.1b. A adjacência  $\mathcal{A}_k$  de  $k$ -vizinhos mais próximos é definida como

$$t \in \mathcal{A}_k(s) \quad \text{se } t \text{ é } k\text{-vizinho mais próximo de } s. \quad (4.3)$$

A relação de adjacência dada pela Equação 4.3 é assimétrica e resulta em relações assimétricas de conectividade definidas na Seção 3.2. Dado que o máximo pode ser platô de densidade, é necessário

garantir conexidade entre as amostras neste máximo através de conexões simétricas. Uma relação assimétrica pode gerar mais de um representante por máximo. Uma forma de definir conexões simétricas no grafo através da extensão da adjacência da Equação 4.3 para uma relação simétrica restrita a um platô dada por

$$\begin{aligned}
 & \text{se } t \in \mathcal{A}_k(s), \\
 & \quad s \notin \mathcal{A}_k(t) \text{ e} \\
 & \rho(s) = \rho(t), \text{ então} \\
 & \mathcal{A}(t) \leftarrow \mathcal{A}_k(t) \cup \{s\}.
 \end{aligned} \tag{4.4}$$

Como resultado, temos um grafo  $k$ -vizinhos mais próximos simétrico em platôs de densidade para um dado valor de  $k$ , cujos pesos dos nós pela Equação 4.1 são ilustrados pela Figura 4.1b. Neste exemplo, dois grupos são identificados na Figura 4.1c e a partir dos máximos de densidade. Através da mudança da função de caminho  $f_{max}$  por  $f_{min}$  no Algoritmo 2, amostras pertencentes às zonas de influência de cada máximo (domo) são rotuladas durante a execução do algoritmo.

### 4.1.2 Estimação do Melhor Grafo de $k$ -vizinhos Mais Próximos

A identificação de raízes é a tarefa principal para eliminar máximos regionais irrelevantes na PDF em casos que o número de grupos é maior que o número desejado (Figuras 4.2b-c).

Diferentes escolhas de  $k$  podem resultar em diferentes florestas de caminhos ótimos, cuja rotulação das árvores representa cortes distintos em  $(\mathcal{N}, \mathcal{A})$ . Medidas de análise de validade de agrupamento podem ser utilizadas, mas geralmente assumem grupos compactos e bem separados [86, 87, 88, 89]. A medida de análise também deve ser independente do formato dos grupos. Neste contexto, Medidas de corte em grafo geralmente são desenvolvidas para separar nós em somente duas partes [28, 29, 30, 31, 32, 33]. No caso de uma floresta, é utilizada a medida de corte em grafo estendida para múltiplos grupos [31]. O melhor valor de  $k$  é aquele que minimiza a medida de corte em um grafo.

Seja  $1/d(s, t)$  os pesos de arcos de um grafo de  $k$ -vizinhos mais próximos  $(\mathcal{N}, \mathcal{A})$ . Algoritmo 3 pode prover em  $L$  um corte em grafo para cada valor de  $k \in [1, (|\mathcal{N}| - 1)]$ . Esse corte é medido por  $C(k)$ .

$$C(k) = \sum_{i=1}^c \frac{W'_i}{W_i + W'_i}, \tag{4.5}$$

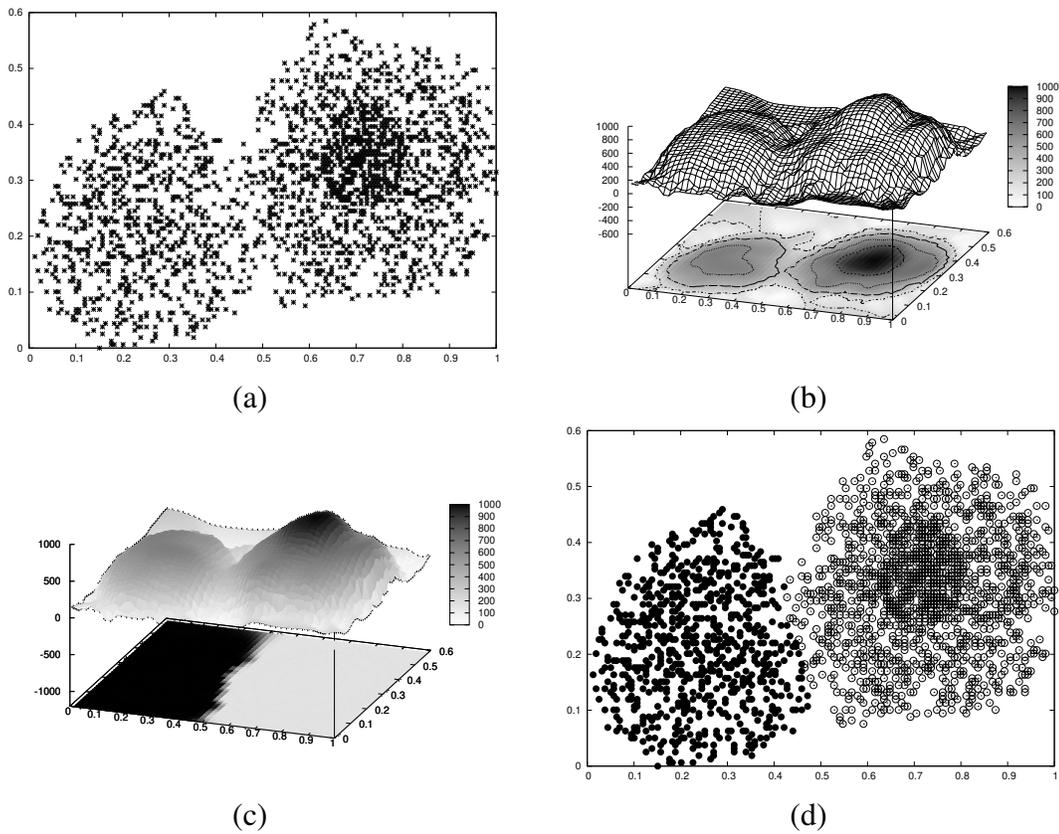


Fig. 4.1: (a) Um espaço bidimensional onde amostras são representadas por pontos. (b) O peso de cada amostra dado pela distribuição de densidade de probabilidade  $\rho(s)$  dado pela Equação 4.1, representada pela topologia da superfície 3D e pelo inverso do brilho de sua projeção 2D no plano abaixo. (c) Os grupos são definidos pelos máximos da PDF, cujas zonas de influência são ilustradas na projeção 2D pelas regiões de cores preta e branca. (d) As amostras recebem o rótulo de seus respectivos máximos.

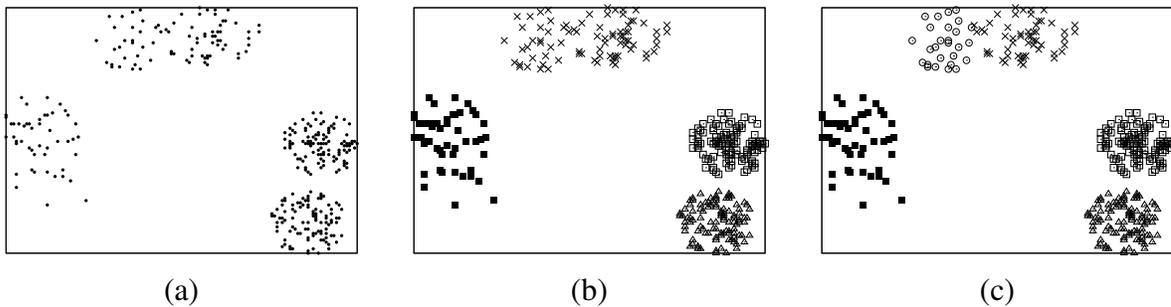


Fig. 4.2: (a) Espaço de característica com uma concentração de amostra por grupo. É possível identificar um número diferente de grupos dependendo da escala: (b) quatro, em uma escala maior, ou (c) cinco grupos em uma escala menor.

$$W_i = \sum_{\forall (s,t) \in \mathcal{A} | L(s)=L(t)=i} \frac{1}{d(s,t)}, \quad (4.6)$$

$$W'_i = \sum_{\forall (s,t) \in \mathcal{A} | L(s)=i, L(t) \neq i} \frac{1}{d(s,t)}, \quad (4.7)$$

$$(4.8)$$

O melhor corte é definido pelo valor mínimo de  $C(k)$ , onde  $W'_i$  considera todos os pesos de arcos entre grupo  $i$  e outros grupos, e  $W_i$  considera todos os pesos de arcos internos dos grupos  $i = 1, 2, \dots, c$ .

A curva  $C(k)$  é geralmente decrescente para valores até um mínimo em algum valor de  $k$ . A partir deste valor a curva passa a ter comportamento crescente. O cálculo exaustivo dentro de todo o intervalo  $k \in [1, (|\mathcal{N}| - 1)]$  é impraticável, mas felizmente desnecessário na maioria das situações porque o mínimo geralmente ocorre para  $k \ll |\mathcal{N}| - 1$ . Também é possível que a curva apresente múltiplos mínimos sendo aquele desejado um mínimo local, porque múltiplas soluções podem existir de acordo com a escala. A solução desejada é o maior  $k$  dentro do intervalo  $[1, k_{max}]$  que possui um mínimo em  $C(k)$ , onde  $1 \leq k_{max} \leq |\mathcal{N}| - 1$ .

### 4.1.3 Zonas de Influência de Máximos

A função de valor de caminho  $f_1$  define uma floresta de caminhos ótimos com raízes em todos os máximos de  $\rho(s)$ .

$$f_1(\langle s_1, s_2, \dots, s_n \rangle) = \min_{\forall s_i, i=2,3,\dots,n} \{H(s_1), \rho(s_i)\} \quad (4.9)$$

para um valor inicial  $H(s_1)$  dado por

$$H(s_1) = \rho(s_1) - \delta, \quad (4.10)$$

$$\delta = \min_{\forall (s,t) \in \mathcal{A} | \rho(t) \neq \rho(s)} |\rho(t) - \rho(s)|.$$

Na prática, um máximo pode ser um componente conexo no grafo  $(\mathcal{N}, \mathcal{A})$ , onde os nós têm o mesmo valor de densidade (pesos). Com o propósito de atribuir um rótulo distinto para cada máximo, apenas uma raiz deve ser escolhida arbitrariamente por cada máximo. Isto pode ser alcançado pela escolha de uma função de valor de caminho  $f_2$ .

$$\begin{aligned}
f_2(\langle s_1 \rangle) &= \begin{cases} \rho(s_1) & \text{se } s_1 \in S \\ H(s_1) & \text{caso contrário.} \end{cases} \\
f_2(\langle s_1, s_2, \dots, s_n \rangle) &= \min_{\forall s_i, i=2,3,\dots,n} \{f_2(\langle s_1 \rangle), \rho(s_i)\}
\end{aligned} \tag{4.11}$$

onde as raízes em  $S$  são identificadas durante a execução do algoritmo e  $H(s_1)$  é dado pela Equação 4.10.

O número de máximos pode ser reduzido se  $\delta$  for aumentado na Equação 4.10 ou pelo cálculo de uma operação morfológica não-extensiva em  $\rho$  tal que  $H(s_1) < \rho(s_1)$ . A escolha do valor de  $\delta$  pode ser facilitada se os valores de  $\rho$  forem escalonados em um intervalo de  $[1, K]$  de números reais, onde  $K$  é uma constante que limita a procura pelo melhor corte. No segundo caso, podem ser utilizados operadores não-extensivos, como uma abertura morfológica ou filtros conexos [80, 21]. Exemplos de tais operadores são aberturas de área e volume. A densidade filtrada é em seguida subtraída de  $\delta$  para garantir  $H(s_1) < \rho(s_1)$ .

O primeiro caso remove colinas de densidade em  $\rho$  com altura abaixo de  $\delta$ . O segundo caso leva em conta outros critérios, como a área mínima ou volume dessas colinas. A Figura 4.3a apresenta um exemplo de um histograma unidimensional onde barras cinzas ilustram uma PDF, em que  $\rho$  é escalonado no intervalo definido por  $K = 10^3$ , com quatro máximos locais. Figura 4.3b mostra as zonas de influência definidas pela floresta de caminhos ótimos (setas) sobre  $V$  (barras brancas) de dois máximos resultantes da filtragem de  $H(t) = \rho(t) - 2$ .

No domínio da imagem, esses filtros usam um conjunto de pixels, uma adjacência espacial e uma imagem nível de cinza [20, 2]. Neste trabalho, substituímos o conjunto de pixels por  $\mathcal{N}$ , usando a relação de  $k$ -vizinhos mais próximos simétrica A (Equação 4.4), e  $\rho$  (Equação 4.1) como função nível de cinza. No entanto, esta relação de adjacência cria um grafo assimétrico, isto é, para cada par de nós  $s$  e  $t$  a existência de arco de  $(s, t)$  não garante um arco  $(t, s)$ . Isto leva a um problema na designação de máximos únicos em platôs de custo se em um mesmo máximo um nó  $s$  tiver um arco para um nó  $t$ , mas o contrário pode não ocorrer. Neste caso o nós  $s$  e  $t$  serão máximos com rótulos diferentes. Isto pode ser resolvido através de uma adjacência simétrica a partir de A, definida na Seção 3.2, o que torna também simétrico o grafo  $(\mathcal{N}, \mathcal{A})$ . Esta solução mostra-se mais elegante do que a união de máximos, que pode ser feita durante ou depois da execução do algoritmo, de raízes que pertencem a um mesmo platô na PDF.

Algoritmo 3 apresenta o algoritmo da IFT modificada para o grafo  $(\mathcal{N}, \mathcal{A})$  e a função valor de caminho  $f_2$ . Ele identifica uma única raiz rotulada para cada máximo que não foi eliminado e calcula caminhos ótimos a partir das raízes em  $P$  seguindo um ordem não-crescente de valor em  $V$ . A saída é um mapa de rótulos  $L$  com o resultado da operação de agrupamento de dados.

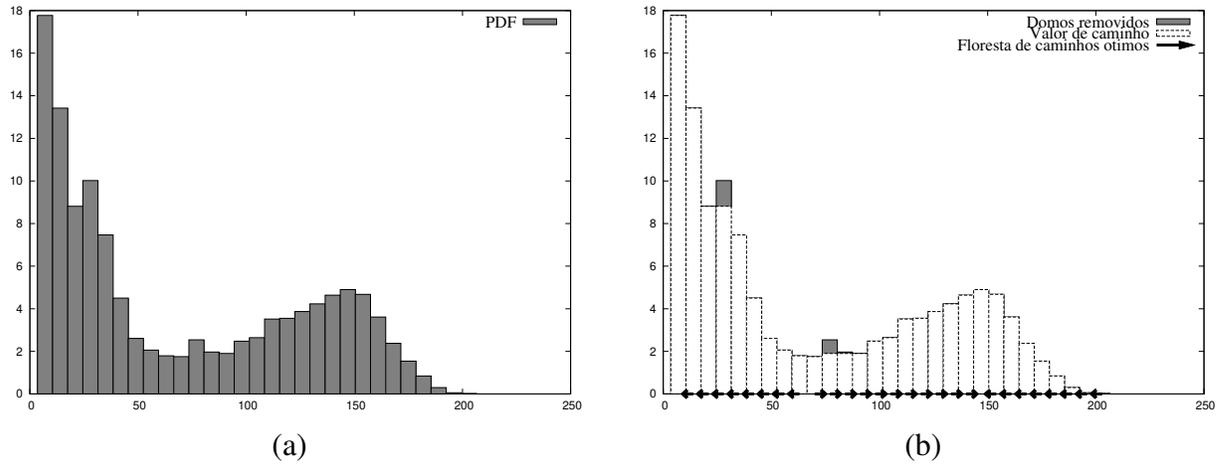


Fig. 4.3: (a) Barras cinzas ilustram um histograma derivado da PDF  $\rho$  unidimensional com quatro máximos locais. (b) A floresta de caminhos ótimos (setas) sobre  $V$  (barras brancas), indicam as zonas de influência de dois máximos restantes depois da filtragem de  $H(t) = \rho(t) - 2$ .

### Algoritmo 3 – AGRUPAMENTO ATRAVÉS DA FLORESTA DE CAMINHOS ÓTIMOS

ENTRADA: Grafo  $(\mathcal{N}, \mathcal{A})$  e funções  $H$  e  $\rho$ , de forma que  $H(s) < \rho(s)$  para todo  $s \in \mathcal{N}$ .

SAÍDA: Mapa de rótulos  $L$ .

AUXILIARES: Mapa de valores de caminhos  $V$ , mapa de predecessores  $P$ , fila de prioridade  $Q$ , variáveis  $tmp$  e  $l \leftarrow 1$ .

1. **Para Cada** amostra  $s \in \mathcal{N}$ , **Faça**  $P(s) \leftarrow nil$ ,  $R(s) \leftarrow s$ ,  $V(s) \leftarrow H(s)$  e *insira*  $s$  em  $Q$ .
2. **Enquanto**  $Q$  não for vazia, **Faça**
3.     *Remova*  $s$  de  $Q$  tal que  $V(s)$  é máximo.
4.     **Se**  $P(s) = nil$  **Então Faça**  $L(s) \leftarrow l$ ,  $l \leftarrow l + 1$ ,  $V(s) \leftarrow \rho(s)$ .
5.     **Para Cada** nó  $t \in \mathcal{A}(s)$ , **Faça**
6.         **Se**  $V(t) < V(s)$ , **Faça**
7.              $tmp \leftarrow \min\{V(s), \rho(t)\}$ .
8.             **Se**  $tmp > V(t)$ , **Então**
9.                 **Faça**  $L(t) \leftarrow L(s)$ ,  $P(t) \leftarrow s$ , e  $V(t) \leftarrow tmp$ . *Atualize a posição de*  $t$  em  $Q$ .

Linha 1 inicializa os mapas e insere todas as amostras em  $Q$ . Um mapa de predecessores  $P$  é utilizado para definir a floresta de caminhos. Cada iteração do laço principal (linhas 2–11), um caminho  $P^*(s)$  de valor ótimo  $V(s)$  é obtido em  $P$  quando a última amostra  $s$  é removida de  $Q$  (linha 3). Empates são resolvidos em  $Q$  através do critério de FIFO. O teste  $P(s) = nil$  na linha 4 identifica  $P^*(s)$  como um caminho trivial  $\langle s \rangle$ . Dado que caminhos ótimos são encontrados em um ordem não-crescente de valor, caminhos triviais indicam nós que estão nos máximos locais. Pela mudança de

$V(s)$  para  $\rho(s)$ , como definida pela Equação 4.11 e indicada na linha 4, o algoritmo faz com que cada nó em cada máximo conquiste o resto dos nós em sua zona de influência. Assim,  $s$  torna-se raiz da floresta na linha 4 com um rótulo distinto que lhe é atribuído. As linhas 5–9 verificam se o caminho alcançado por nó adjacente  $t$  através de  $s$  é melhor que o caminho atual com sufixo  $t$  e atualiza a fila  $Q$ ,  $V(t)$ ,  $L(t)$  e  $P(t)$ . Se ocorressem duas ou mais raízes em um mesmo platô, simetricamente conexas, o caminho entre as raízes teria um mesmo valor. Essa hipótese não é possível, pois uma das raízes recebe o valor de  $\rho(s)$  enquanto os nós restantes no platô são dominados por esta com valor de densidade  $H(s) < \rho(s)$ .

O Algoritmo 3 executa em  $\Theta(|\mathcal{A}| + |\mathcal{N}| \log |\mathcal{N}|)$  se  $Q$  é uma estrutura de dados *heap* balanceada [19]. Essa complexidade pode ser reduzida a  $\Theta(|\mathcal{A}| + |\mathcal{N}|K)$  se  $\rho$  e  $h$  forem convertidos para valores inteiros no intervalo de  $[0, K]$  e  $Q$  for implementada com *bucket sorting* [90].

#### 4.1.4 Zonas de Influência de Sementes

Um conjunto de sementes  $\mathcal{S} \subset \mathcal{N}$  pode juntar zonas de influência de alguns máximos, reduzindo o número de grupos. O conjunto  $\mathcal{S}$  pode ser definido por limiarização na PDF:  $s \in \mathcal{S}$  se  $\rho(s) > T$ , para um limiar  $1 < T < K$  e  $1 \leq \rho(s) \leq K$ . Os componentes conexas em  $(\mathcal{N}, \mathcal{A})$ , que estão acima de  $T$ , são rotulados com números inteiro consecutivos de 1 até  $c$  de modo a obter  $c$  grupos. Neste caso, cada grupo é uma floresta cujas árvores de caminhos ótimos são enraizadas nas amostra do respectivo componente conexo.

Seja a função  $\Lambda$  a função de rotulamento, isto é,  $\Lambda(s) = \Lambda(t)$  quando  $s$  e  $t$  pertencem a um mesmo componente conexo, e  $\Lambda(s) \neq \Lambda(t)$  caso contrário. O algoritmo de OPF propagará os rótulos dos componentes conexas para os nós restantes quando a função de valor de caminho é definida como

$$\begin{aligned} f_3(\langle s_1 \rangle) &= \begin{cases} H(s_1) & \text{se } s_1 \in \mathcal{S} \\ -\infty & \text{caso contrário.} \end{cases} \\ f_3(\langle s_1, s_2, \dots, s_n \rangle) &= \min_{\forall s_i, i=2,3,\dots,n} \{H(s_1), \rho(s_i)\} \end{aligned} \quad (4.12)$$

onde  $H(s_1) = K$ . Se  $H(s_1) < \rho(s_1)$  então as zonas de influência de alguns componentes conexas podem desaparecer. Isso pode ser desejável em alguns casos, como será discutido no Capítulo 5. Algoritmo 4 apresenta o algoritmo de IFT modificado para grafo  $(\mathcal{N}, \mathcal{A})$  e função de valor de caminho  $f_3$ .

**Algoritmo 4** – AGRUPAMENTO ATRAVÉS DA FLORESTA DE CAMINHOS ÓTIMOS COM IMPOSIÇÃO DE MARCADORES

ENTRADA: Grafo  $(\mathcal{N}, \mathcal{A})$  e funções  $h, \Lambda, S$  e  $\rho$ , de forma que  $H(s) < \rho(s)$  para todo  $s \in \mathcal{N}$ .  
 SAÍDA: Mapa de rótulos  $L$ .  
 AUXILIARES: Mapa de valores de caminhos  $V$ , mapa de predecessores  $P$ , fila de prioridade  $Q$  e variável  $tmp$ .

1. **Para Cada** amostra  $s \in \mathcal{N} \mid S$ , **Faça**  $V(s) \leftarrow -\infty$ .
2. **Para Cada** amostra  $s \in S$ , **Faça**  $P(s) \leftarrow nil$ ,  $L(s) \leftarrow \Lambda(s)$ ,  $V(s) \leftarrow H(s)$ , e insira  $s$  em  $Q$ .
3. **Enquanto**  $Q$  não for vazia, **Faça**
4.     Remova  $s$  de  $Q$  tal que  $V(s)$  é máximo.
5.     **Para Cada** nó  $t \in \mathcal{A}(s)$ , **Faça**
6.         **Se**  $V(t) < V(s)$ , **Faça**
7.              $tmp \leftarrow \min\{V(s), \rho(t)\}$ .
8.             **Se**  $tmp > V(t)$ , **Então**
9.                 **Se**  $V(t) \neq -\infty$  **Então** remova  $t$  de  $Q$ .
10.                 **Faça**  $L(t) \leftarrow L(s)$ ,  $P(t) \leftarrow s$ , e  $V(t) \leftarrow tmp$ . Insira  $t$  em  $Q$ .

A maior parte dos comentários para Algoritmo 3 são aplicados para Algoritmo 4. Eles diferem essencialmente nos seguintes aspectos. Todas as sementes  $s \in S$  virarão raízes de uma floresta de caminhos ótimos  $P$ , quando  $H(s) = K$ . Os rótulos destas sementes são propagados em  $L$  durante o algoritmo. Linha 2 insere somente sementes em  $Q$ . Logo, quando o teste da linha 6 é verdadeiro,  $t$  pode estar ou não em  $Q$ . O teste da linha 9 identifica quando o nó  $t$  está em  $Q$  e o remove, atualiza os mapas e insere novamente em  $Q$  em uma melhor posição (linha 10). Se  $t$  nunca for alcançado por nenhum caminho, então  $V(t) = -\infty$ . Neste caso, linhas 9 e 10 inicializam os mapas para  $t$  e o inserem em  $Q$  pela primeira vez. Amostras removidas de  $Q$  na linha 4 nunca irão retornar para  $Q$ .

## 4.2 Extensão para Grandes Bases de Dados

A complexidade computacional do Algoritmo 3 é  $O(k|\mathcal{N}| + |\mathcal{N}| \log \mathcal{N})$  quando  $Q$  é uma *heap* binária, e a estimação do melhor  $k$  requer seu cálculo inúmeras vezes. Isto pode demorar alguns minutos em computadores modernos para  $|\mathcal{N}| > 1000$ . Esse problema torna-se quase impraticável para imagens 2D/3D. Em [91], o número de arcos é consideravelmente reduzido pelo definição de  $\mathcal{A}$  como  $(s, t) \in \mathcal{A}$  como  $\|t - s\| \leq h_i$  e  $d(s, t) \leq h_f$  para segmentação de imagem. O resultado, no entanto, torna-se um compromisso entre a escolha de  $h_i$ , cujos menores valores aumentam o número de grupos, e a escolha de  $f(\langle t \rangle)$ , que pode reduzir esse número. Descartando toda restrição no domínio da imagem, o método pode ser implementado em grafos de  $k$ -vizinhos mais próximos de forma incremental.

A extensão é baseada na seleção de um subconjunto  $\mathcal{N}' \subset \mathcal{N}$ . Sejam  $V$  e  $L$  os mapas de caminhos ótimos e rótulos, respectivamente, do Algoritmo 3 calculados com o melhor grafo de  $k$ -vizinhos mais próximos  $(\mathcal{N}', \mathcal{A})$ . Um amostra  $t \in \mathcal{N} \setminus \mathcal{N}'$  pode ser classificada em um dos grupos pela identificação de qual raiz lhe ofereceria um caminho ótimo como se fizesse parte da floresta. Considerando os  $k$ -vizinhos mais próximos de  $t$  em  $\mathcal{N}$ , a Equação 4.1 é usada para calcular os caminhos ótimos  $\pi_s \cdot \langle s, t \rangle$ , e para selecionar aquela que satisfaz

$$V(t) = \max_{\forall (s,t) \in \mathcal{A}} \{\min\{V(s), \rho(t)\}\} \quad (4.13)$$

Seja o nó  $s^* \in \mathcal{N}'$  aquele que satisfaz a Equação 4.13. A classificação simplesmente atribui  $L(s^*)$  como grupo de  $t$ .

### 4.3 Aplicação em Segmentação de Imagem/Vídeo

Uma imagem pode ser considerada um par  $(\mathcal{I}, \vec{I})$  em que  $\mathcal{I} \subset Z^2$  é o domínio da imagem e  $\vec{I}(s) = \{I_1(s), I_2(s), \dots, I_m(s)\}$  é uma função vetorial, a qual atribui um conjunto de propriedades da imagem a cada pixel  $s \in \mathcal{I}$ . Por exemplo,  $\{I_1(s), I_2(s), I_3(s)\}$  podem ser os valores vermelho, verde e azul de  $s$  em uma imagem colorida. Pode ser definido  $\vec{I}(s) = \mathbf{x}(s)$  ou calculados vetores de características baseados em alguma transformação aplicada a  $\hat{I}$ . Alguns resultados interessantes foram obtidos com filtragem multi-escala (Seção 4.3.1).

O grafo  $(\mathcal{N}, \mathcal{A})$  definido pelo conjunto de pixels  $\mathcal{I} = \mathcal{N}$  e uma relação de adjacência  $\mathcal{A}$  deve ser definido de tal forma que grupos de pixels com propriedades similares formem objetos relevantes ou ao menos dividam estes objetos em poucas regiões, que pode ser facilmente agrupadas posteriormente. No entanto, objetos distintos que têm propriedade similares e a cardinalidade de  $\mathcal{I}$  proíbe grafos densos devido ao seu excessivo custo computacional. Como consequência disto, diversas técnicas geralmente consideram a relação de adjacência com restrições de conexidade no domínio espacial [31, 33, 8]. Outra abordagem é diminuir a resolução espacial da imagem através de sub-amostragem. O número de amostras deve ser suficiente para representar os objetos de acordo com sua escala na imagem. Em ambos os casos, a escolha da característica correta é a chave para a conseguir um bom resultado em qualquer conjunto de dados. Neste contexto, características que envolvem uma vizinhança ao redor do pixel com diversos graus de escala (multi-escala) são as que apresentam melhores resultados na literatura [92].

### 4.3.1 Características Multi-Escalas

Considere a função Gaussiana  $G_\sigma$  com desvio padrão  $\sigma$ . As convoluções entre  $G_\sigma$  e cada componente da imagem  $I_i$ ,  $i = 1, 2, \dots, m$  para  $S$  valores crescente de  $\sigma$  criam um vetor de característica  $\mathbf{x}(s)$  com  $mS$  valores de características para cada pixel  $s \in \mathcal{I}$ . Essa idéia segue o procedimento padrão de filtragem multi-escala [92]. Outra idéia é usar convoluções com um kernel Laplaciano  $\sigma^2 \nabla^2 G$  [93]. Ambas as abordagens apresentam bons resultados com baixos valores de  $\sigma$ , mas o deslocamento da borda do objeto fica perceptível para valores crescentes de escala devido a filtragem linear (Figuras 4.4a e 4.4b).

Por outro lado, operadores conexos podem simplificar imagens sem criar falsas bordas [80, 21, 20]. Filtros sequenciais alternados (*Alternate Sequential Filters — ASF*) por reconstrução podem, por exemplo, ser utilizados na criação de características multi-escalas sem o deslocamento de bordas [94]. Um fechamento por reconstrução seguido por uma abertura por reconstrução são utilizados como ASF neste trabalho. O elemento estruturante é um disco com raio  $r$  e com escalas que são criadas pela variação de  $r = 1, 2, \dots, S$ . O filtro é aplicado para cada componente de imagem criando  $mS$  valores de características para cada pixel. O deslocamento de borda não ocorre nem para grandes escalas (Figuras 4.4c e 4.4d).

### Pesos Baseados no Gradiente da Imagem

Considere o grafo  $(\mathcal{N}, A)$  como uma discretização de uma imagem  $\hat{I} = (\mathcal{N}, G_b)$ , cujos pesos dos arcos  $w_i$  são dados pela Equação 4.15 através do cálculo da magnitude do gradiente  $G_b$  na imagem em nível de cinza ou colorida, onde o número de bandas da imagem é dado por  $b = 1$  ou  $b = 3$ , respectivamente.

$$t \in \mathcal{A}_8(s) \quad \text{se} \quad \|t - s\| \leq \sqrt{2}, \quad (4.14)$$

$$w_i(s, t) = \max_{b=1, \dots, 3} \left\{ \left| \frac{\vec{G}_b(s) + \vec{G}_b(t)}{2} \right| \right\}. \quad (4.15)$$

As características multi-escala da imagem também são utilizadas para o cálculo do peso atribuído aos arcos (Equação 4.15) do grafo baseado no gradiente da imagem  $G_i$  em ambas transformadas IFT-*watershed*, de marcadores de nível de cinza [2] e de marcadores rotulados [50]. A imagem gradiente  $(\mathcal{N}, G_i)$  é calculada usando a adjacência de vizinhos-8 da Equação 4.14. Figuras 4.7a-d ilustram o gradiente das respectivas imagens nas Figuras 4.5a-d usando a Equação 4.17.

Os pesos dos arcos são baseados na magnitude do gradiente multi-escala da imagem  $\vec{G}_i(s)$  são definidos pela Equação 4.16.

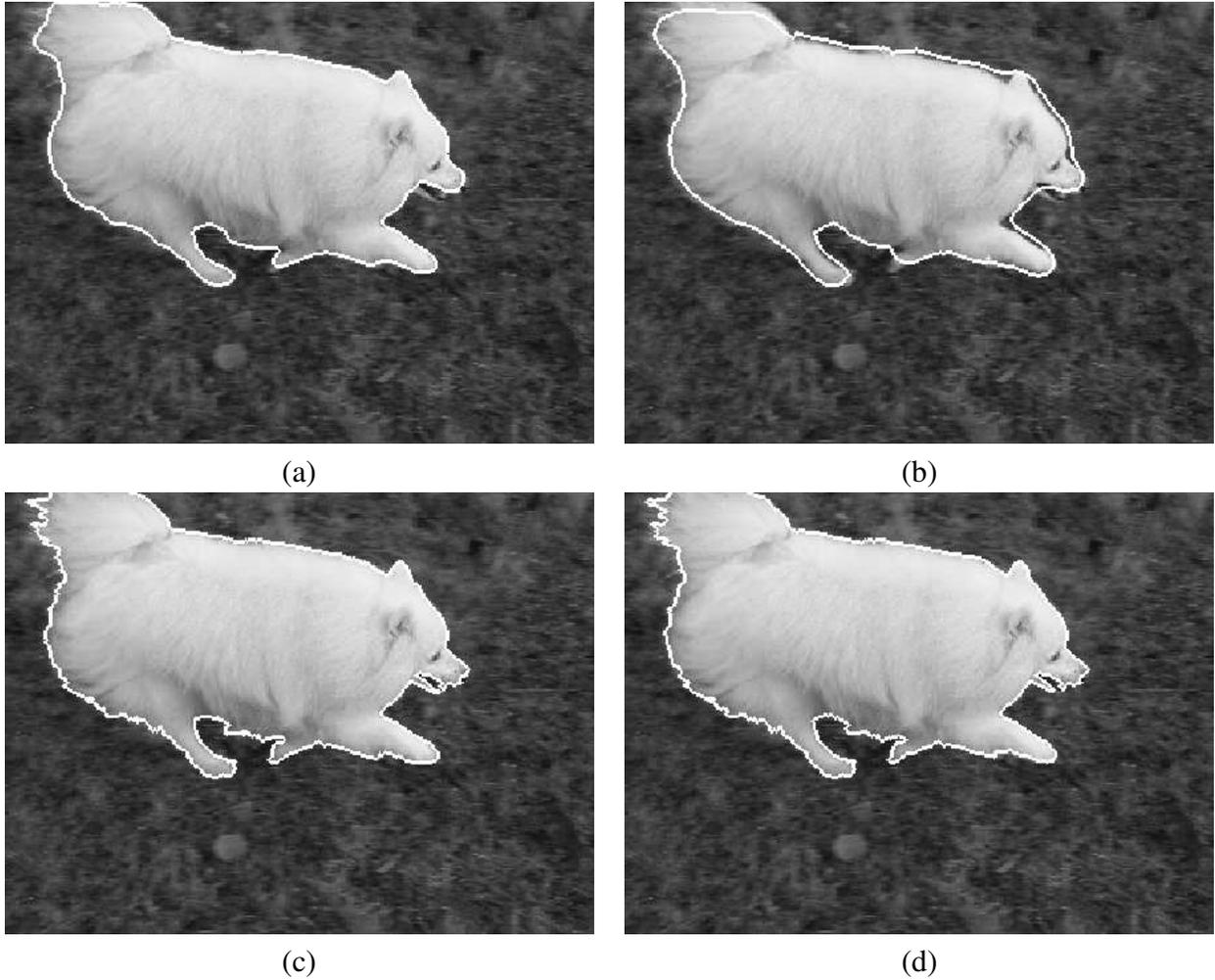


Fig. 4.4: O interior e exterior do contorno sobre a imagem de entrada nível de cinza indicam dois grupos. Estes resultados são obtidos com o Algoritmo 3,  $d_i = 5$ ,  $h$  criado a partir da eliminação de domos de  $\rho$  com área menor que  $10 \cdot 10^3$  pixels, e vetores de características multi-escala criados por: (a) filtro Laplaciano com 5 escalas, (b) filtro Laplaciano com 15 escalas, (c) ASF por reconstrução com 5 escalas, e (d) ASF por reconstrução com 15 escalas. Neste exemplo, a largura de banda  $d_f$  foi calculada pela Equação 4.1 para o melhor grafo  $k$ -vizinhos mais próximos usando sub-amostragem de 16 para 1.

$$\vec{G}_i(s) = \sum_{j=1}^S \sum_{\forall t \in \mathcal{A}_8(s)} [v_{i,j}(t) - v_{i,j}(s)] \vec{s}t, \quad (4.16)$$

$$G(s) = \max_{i=1, \dots, m} \|\vec{G}_i(s)\| \quad (4.17)$$

onde  $v_i$  é a  $i$ -ésimo valor de característica de  $\mathbf{x}$ ,  $\mathcal{A}_8(s)$  é a relação de adjacência que resulta em

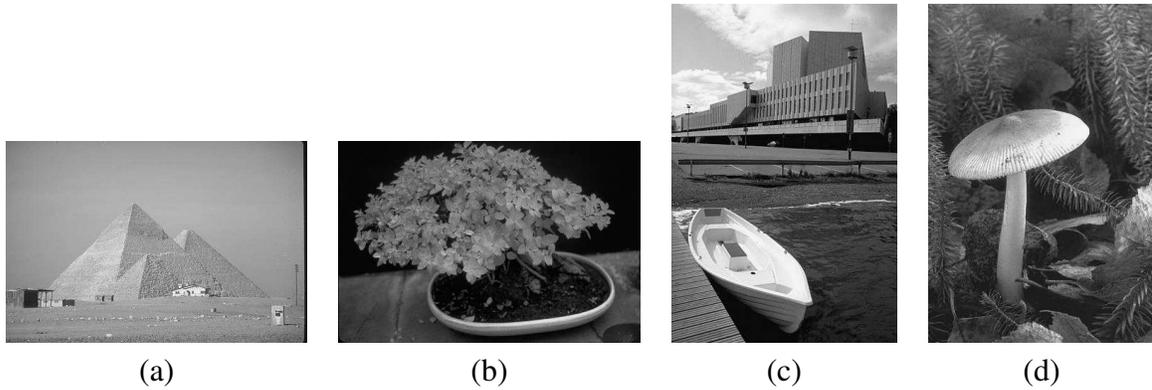


Fig. 4.5: (a-d) Imagens coloridas de cenas naturais apresentadas em nível de cinza.

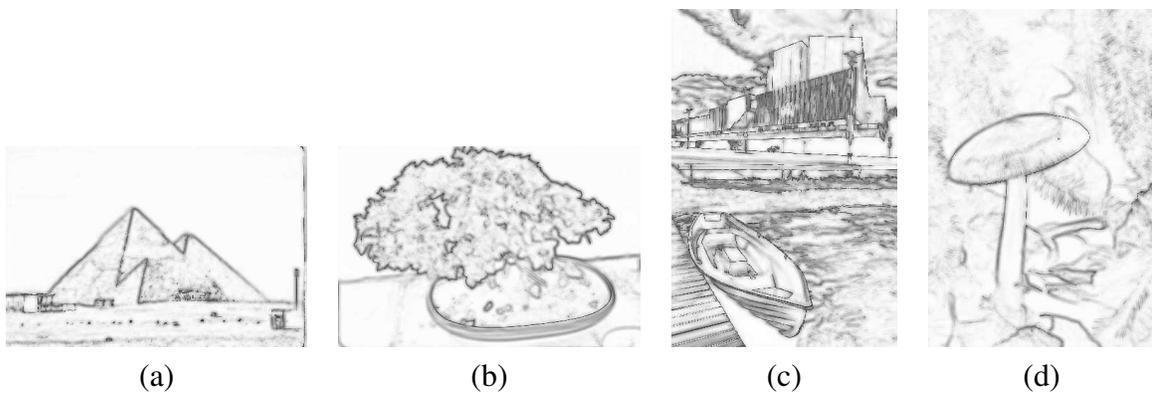


Fig. 4.6: (a-d) Imagens de densidade  $\rho$  calculadas nas respectivas imagens das Figuras 4.5a-d usando Equação 4.1, onde um maior o valor de brilho indica um maior valor de densidade na PDF.

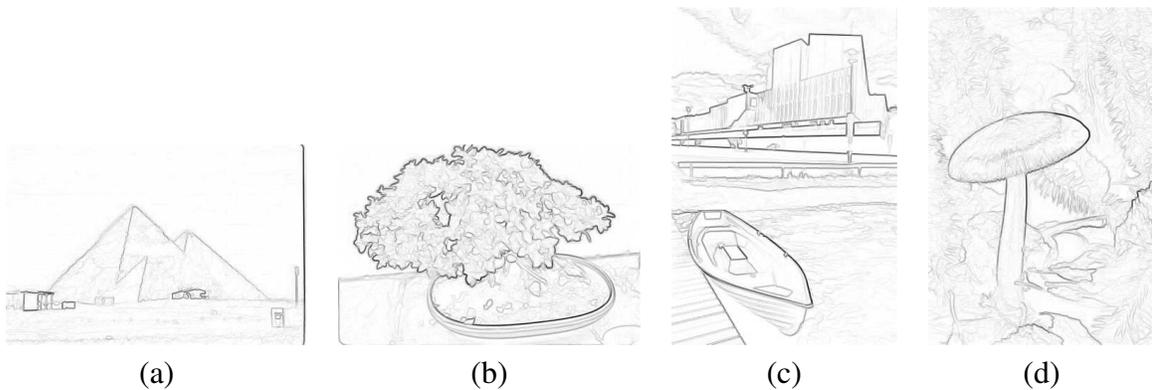


Fig. 4.7: (a-d) Imagens de gradiente calculadas nas respectivas imagens das Figuras 4.5a-d usando Equação 4.17, onde um menor o valor de brilho indica um maior valor de gradiente.

um conjunto de pixels vizinhos-8 de  $s$ , e  $\vec{st}$  é o vetor que conecta  $s$  a  $t$  no domínio espacial. Esta formulação tenta capturar a projeção dos vetores de gradiente em mais de dois eixos e em diferentes escalas.

### Pesos Baseados no Objeto

Sejam o vetor de características multi-escala  $\vec{v}(s)$  e  $S$  o conjunto semente de pixels representativos da imagem dentro de cada objeto  $l = 1, 2, \dots, c$ , a probabilidade de um nó ser do objeto esperado,  $\mathcal{P}(l | s)$ , é proporcional às probabilidades condicional,  $\mathcal{P}(s | l)$ , e *a priori*,  $\rho(l)$ , pela Regra de Bayes (Seção 2.5) na Equação 2.19 utilizada para amostras do grafo (Equação 4.18).

$$\mathcal{P}(l | s) = \frac{\mathcal{P}(s | l)\rho(l)}{\rho(s)} \quad (4.18)$$

Para múltiplos objetos, o conjunto semente deve ser escolhido dentro de regiões do objeto  $i$  da imagem formada até que o valor de  $\mathcal{P}(l = i | s)$ ,  $i = 1, 2, \dots, c$ , seja superior aos valores das regiões restantes. Um exemplo é ilustrado na Figura 4.8a para o caso de somente de objeto e fundo da imagem (Figura 4.8b). Na Figura 4.8c a estimação de sementes é melhorada e os valores de brilho do objeto (claro) na imagem da Figura 4.8d ficam mais distintos dos valores de fundo (escuro).

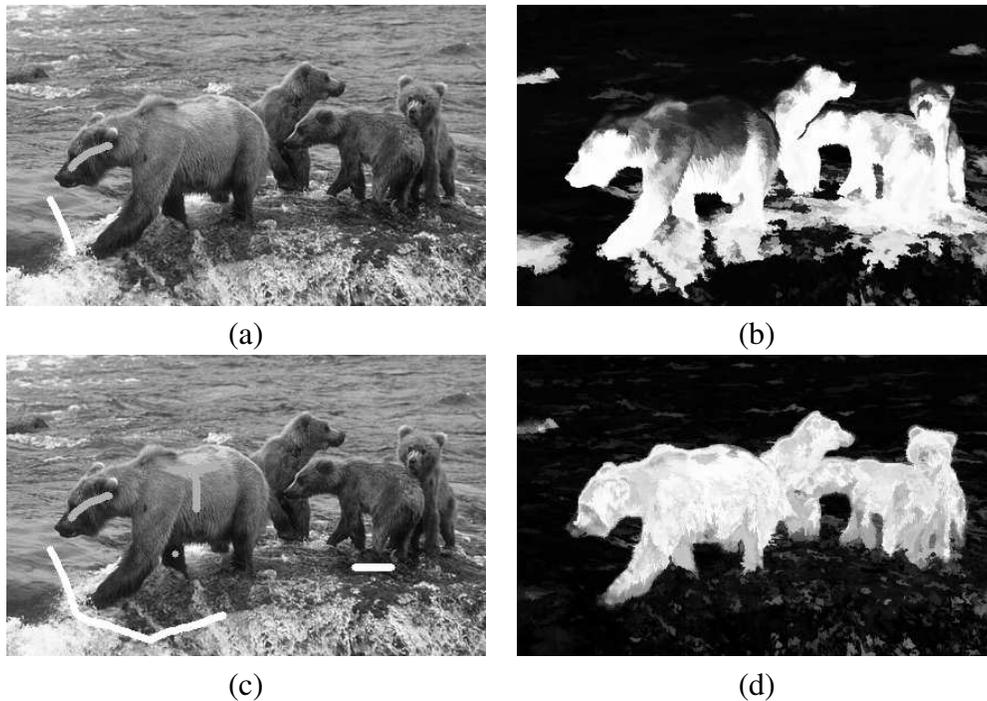


Fig. 4.8: Imagem onde o objeto é a família de ursos. (a-b) Um conjunto semente inicial e a imagem correspondente  $\rho(l = 1 | s)$ . (c-d) Uma melhor estimação onde sementes internas e externas tentam separar características do objeto são adicionadas em  $\rho(l = 1 | s)$ .

As descontinuidades entre cada objeto  $l$  e o restante da imagem é capturada por vetor de gradiente  $\vec{G}_l(s)$ , definido como

$$\vec{G}_l(s) = \sum_{\forall t \in \mathcal{A}_8(s)} [\rho(l | t) - \rho(l | s)] \vec{st} \quad (4.19)$$

onde  $\vec{st}$  é o vetor unitário conectando  $s$  e  $t$  no domínio da imagem. A adjacência Euclidiana 2D  $\mathcal{A}_8$  (Equação 4.14) é utilizada na estimação do vetor gradiente  $\vec{G}_l(s)$  usando a soma das primeiras derivadas de  $\mathcal{P}(l | s)$  nas 8 direções. Para cada arco  $(s, t) \in \mathcal{A}_8$ , a magnitude do vetor gradiente médio é utilizado como peso  $w_{o,l}(s, t)$  com respeito ao objeto  $l$ . O peso final é calculado pelo máximo  $w_{o,l}(s, t)$  entre todos os objetos, seguindo a idéia da função discriminante de Bayes (Seção 2.5.1).

$$w_{o,l}(s, t) = \left| \frac{\vec{G}_l(s) + \vec{G}_l(t)}{2} \right| \quad (4.20)$$

$$w_o(s, t) = \max_{l=1,2,\dots,c} \{w_{o,l}(s, t)\}. \quad (4.21)$$

### 4.3.2 Redução de Resolução Espacial

Foi mostrada na Seção 4.1.2 como achar o melhor grafo  $k$ -vizinhos mais próximos de acordo com a Equação 4.5. Logo,  $d_f$  é obtida pela Equação 4.2. Dado o custo computacional excessivo que proíbe considerar um grafo  $k$ -vizinhos mais próximos cujos nós são todos pixels em  $\mathcal{I}$ , um conjunto de pixels é sub-amostrado e somente considerado um grafo  $k$ -vizinhos mais próximos  $(\mathcal{N}', \mathcal{A}')$ , onde  $\mathcal{N}' \subset \mathcal{I}$  e  $\mathcal{A}'$  é dada pela Equação 4.4 sobre as amostras de  $\mathcal{N}'$ . O conjunto  $\mathcal{N}'$  pode ser definido pela sub-amostragem de nós e vetores de características de  $\mathcal{I}$  de 4 para 1, 8 para 1, ou 16 para 1, horizontalmente e verticalmente, de forma que as amostras ainda representam todos os objetos (Figura 4.9a). A amostragem pode também ser aleatória no espaço da imagem, desde que o número de amostras de cada objeto seja balanceado, isto é, todos os objetos contém um número suficiente de amostras que o representa no domínio espectral. Para tanto, uma heurística ou informação *a priori* sobre o conjunto de dados deve ser aplicada. Isto é essencial para a estimação de densidade se aproxime da PDF original da imagem, onde todos os máximos são bem definidos, ou seja, não apresentam máximos irrelevantes provenientes de picos de alta frequências.

A estimação da PDF utiliza um novo grafo formado pelos nós do sub-grafo  $t \in \mathcal{N}'$  e os demais nós  $s \in \mathcal{N} | \mathcal{N}'$  com arcos  $(s, t)$  conectando ambos os sub-grafo constituintes. A relação de adjacência  $\mathcal{A}'$  dos arcos é formada pela Equação 4.4 no conjunto  $\mathcal{N} | \mathcal{N}'$ .

A densidade calculada em cada nó dos pixels restantes em  $s \in \mathcal{N} | \mathcal{N}'$  possui uma informação global do conjunto de dados, pois a PDF é estimada nas sub-amostras da imagem. Os nós podem estar desconexos no domínio da imagem mas próximos no domínio multidimensional. A Figura 4.9b

ilustra a distribuição de pesos e dos máximos, criados pela sub-amostragem da Figura 4.9a. Os máximos marcados com “×”, na topologia do grafo  $\mathcal{N}'$  não são máximos no domínio da imagem.

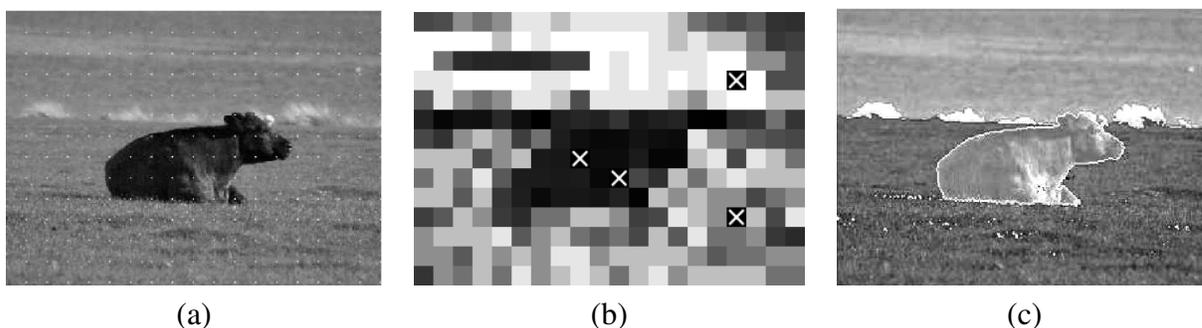


Fig. 4.9: (a) A sub-amostragem da imagem a cada 16 pixels, verticalmente e horizontalmente, é ilustrada por marcadores de cor branca. (b) Os valores de densidade são representados por valores de brilho (equalizados) da imagem sub-amostrada em (a) os máximos do sub-grafo gerado são marcados com “×”. (c) Os demais pixels da imagem em (a) são rotulados com os rótulos das raízes do sub-grafo em (b) que lhes ofereceriam um caminho ótimo se fizessem parte do sub-grafo.

A classificação dos demais pixels no grafo  $\mathcal{N} \mid \mathcal{N}'$  é feita através da atribuição dos mesmos rótulos das raízes no sub-grafo  $\mathcal{N}'$  que oferecem um caminho ótimo, como se esses pixels fizessem parte do grafo dado pela relação de adjacência  $\mathcal{A}'$ . Através da classificação de pixels, a imagem será rotulada com o mesmo número de grupos presentes no sub-grafo. Como não existe uma restrição de conectividade espacial, os rótulos podem estar desconexos no domínio da imagem, como ilustra o exemplo da Figura 4.9c. Caso seja necessário, um processamento posterior pode rotular com restrição de conectividade espacial os grupos resultantes da classificação. Em algumas aplicações, no entanto, essa restrição não é desejável, como sensoriamento remoto e segmentação de vídeo, pois nestes casos um mesmo objeto pode estar naturalmente desconexo. Comparações com outros métodos de classificação são apresentadas no Capítulo 5.

### 4.3.3 Restrição de Conectividade Espacial

O cálculo de  $k$ -vizinhos mais próximos envolve uma ordenação da distância Euclidiana de todos os nós do grafo para cada o cálculo da adjacência de um nó do grafo. Abordagens na literatura utilizam técnicas para acelerar o cálculo da distância, como o uso de busca de distância parcial (*partial distance search* — PDS) ou pré-calculam uma tabela de distância entre todos os vetores de características [95]. Devido a alta correlação espacial da imagem, apenas nós dentro em uma vizinhança podem ser considerados. A adjacência  $\mathcal{A}$  pode ser definida como

$$t \in \mathcal{A}(s) \quad \text{Se } d(s, t) \leq d_f \text{ e } \|t - s\| \leq d_i, \quad (4.22)$$

onde  $d_i$  é a máxima distância Euclidiana entre pixels  $s$  e  $t$  no domínio da imagem. A probabilidade da função de densidade  $\rho$  é calculada pela Equação 4.1, e Algoritmos 3 e 4 podem ser diretamente aplicados com o grafo resultante da imagem  $(\mathcal{I}, \mathcal{A})$ . Maiores valores de  $d_i$  provem mais nós para uma melhor estimacão da PDF (com menos máximos irrelevantes), mas somente isso conecta objetos distintos. Uma boa estimacão de  $d_f$  é importante para evitar essa conexão. O valor de  $d_f$  é estimado pelo método apresentado na Seção 4.1.2. Esta abordagem tenta encontrar o  $d_f$  global da imagem que melhor separa os objetos da imagem em uma dada escala.

A estimacão da PDF com restrição espacial apresenta um grande número de máximos locais irrelevantes. No entanto, o aumento de  $d_i$  eleva o custo computacional e torna menos precisa a separação espacial entre regiões devido a suavização excessiva nas bacias de valor de caminho. A filtragem por  $h$  é uma solução para eliminar o número excessivo de máximos (Figuras 4.10a-b) sem o aumento de  $d_i$ . No entanto, em alguns casos as regiões podem ser unidas de maneira indesejável. Em alguns trechos, as bordas das regiões não acompanham a borda do objeto e o processo de achar um valor de filtragem pode não resolver o problema. A Figura 4.10a ilustra um exemplo onde a uma imagem super-segmentada com um grande número de máximos com  $k = 29$  vizinhos mais próximos, mas a borda do objeto (vaca) é delineada. Figuras 4.10b-c mostram, respectivamente, a PDF de cada pixel da imagem e o resultado da filtragem por abertura de volume de  $30 \cdot 10^3$  pixels. Neste caso, a borda do objeto não é delineada corretamente, pois nesse trecho da imagem os máximos da PDF referente ao objeto foram filtrados e se juntam ao máximo da região de fundo.

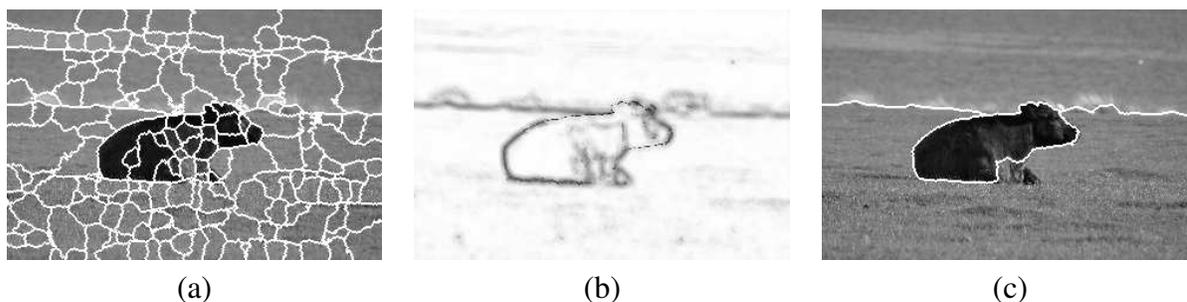


Fig. 4.10: (a) O número de máximos na imagem, encontrado com  $k = 29$  vizinhos,  $d_i = 5$  e  $d_f = 1.705083$ , é representado pelo número de regiões que segmentam a imagem. (b) Os valores de densidade estimada com restrição espacial são representados por valores de brilho na imagem. (c) Um grande número de máximos em (a) é filtrado por uma filtragem por abertura de volume de  $30 \cdot 10^3$  pixels.

#### 4.3.4 Segmentação Interativa de Imagens

Em muitos problemas de reconhecimento de padrões, processamento de imagem, visão computacional e computação gráfica, o desenvolvimento de soluções totalmente automáticas é uma tarefa muito difícil ou quase impossível. Apesar do esforço em pesquisa, uma solução totalmente automática para o problema de segmentação de imagem, ainda é uma meta inatingível. Outras tarefas podem incluir recuperação de imagens por conteúdo (CBIR), onde é geralmente muito difícil recuperar imagens que casam com a expectativa do usuário. Seres humanos têm uma incrível habilidade em distinguir regiões diferentes da imagem ou separar diferentes classes de objetos. Mais importante, as intenções do usuário podem diferir em vários cenários de aplicações. Além disso, em muitas aplicações como imageamento médico, diagnóstico assistido por computador e em outras aplicações de missão crítica, soluções totalmente automáticas podem ser indesejadas e a intervenção e participação humanas no processo de decisões podem ser necessárias. Logo, é tanto útil e algumas vezes necessário incorporar o conhecimento de alto nível e intenções humanas nos algoritmos computacionais para solucionar problemas em imageamento, visão e reconhecimento de padrões. Abordagens interativas provêm soluções semi-automáticas. Os usuários estão dentro do ciclo computacional e interativamente entram com seu conhecimento e intenções que são tomadas como restrições e prioridades pelos algoritmos computacionais para iterativamente refinar os modelos e soluções. Para muitos problemas em reconhecimento de padrões, processamento de imagem, visão computacional, e computação gráfica, tais abordagens interativas podem oferecer um paradigma de uma solução mais realista. Estas abordagens estão dentro de contexto de classificação semi-supervisionadas, onde uma informação indireta é dada sobre o conjunto de amostras.

#### Floresta de Caminhos Ótimos de Marcadores

O uso de marcadores no Algoritmo 4 pode juntar, com mesmo rótulo, zonas de influência de máximos diferentes. Marcadores podem ser obtidos por limiarização [9], onde o conjunto semente consiste de todos os nós que possuem valor de densidade acima de um valor estipulado pelo usuário ou através de alguma heurística. A Figura 4.11a mostra os conjuntos sementes obtidos por uma limiarização na PDF em 95% do seu valor máximo. O resultado desse procedimento é uma abordagem semi-automática, mais apropriada para aplicações práticas onde o limiar pode ser calibrado. A determinação do valor de corte para obtenção dos rótulos não é intuitivo para o usuário e pode gerar sementes em lugares indesejáveis.

Se os marcadores são diretamente impostos pelo usuário através de uma interface gráfica, os marcadores podem não refletir o interesse do usuário, pois os máximos não são visíveis dado que sua conexão vai além do domínio da imagem. A Figura 4.11c mostra o resultado do delineamento, onde

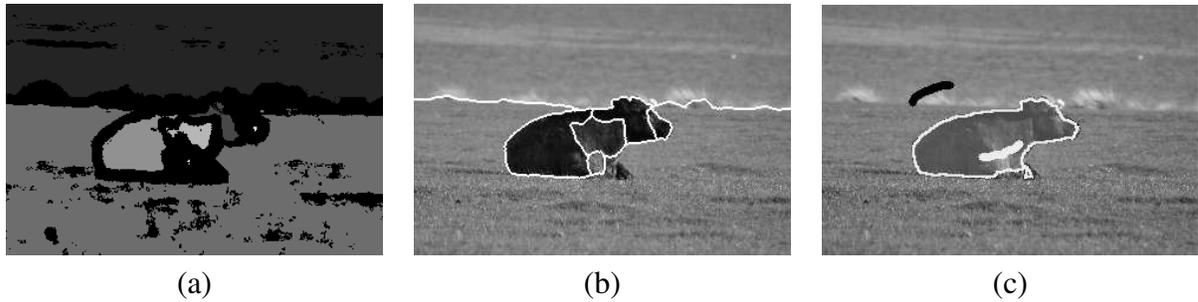


Fig. 4.11: (a) Uma limiarização na PDF em 95% do seu valor máximo gera componentes que são rotuladas de acordo com sua conectividade na imagem que servem de sementes para (b) o Algoritmo 4, que une os máximos com mesmos rótulos. (c) Os marcadores de objeto (branco) e fundo (preto) são impostos pelo usuário, mas o processo não é intuitivo.

somente as regiões com rótulo de objeto (branco) são unidas.

Uma outra abordagem é prover ao usuário um mecanismo de correção da delimitação, combinando a abordagem semelhante a *watershed* de marcadores (WMT) com o uso da IFT. O algoritmo de OPF gera partições da imagens que podem ser unidas com marcadores, ou corrigidas utilizando o algoritmo IFT para correção apenas nas regiões que envolvem os marcadores.

Comumente bordas internas indesejáveis podem interferir na distribuição de pesos e somente pelo cálculo da Equação 4.17 não possibilita ao usuário ter controle de quais regiões deverão apresentar uma distribuição pesos de pequeno valor. Tal procedimento requer treinamento de quais características devem apresentar baixo gradiente. Através do uso de marcadores é possível estimar o gradiente da probabilidade condicional de um objeto (Equações 2.14 e 2.20), que possui altos valores na borda do objeto e baixos valores no restante da imagem. Muito embora um classificador Bayesiano seja supervisionado, o algoritmo OPF pode ser usado para encontrar o valor de  $k$ -vizinhos e para definir os rótulos dos grupos no conjunto de treinamento. A classificação Bayesiana das amostras dos marcadores pode ser feita usando as probabilidades *a priori* do objeto e as densidades condicionais da Equação 2.20. Para  $c$  classes de objetos  $l = 1, 2, \dots, c$ , incluindo o objeto de fundo, sem perda de generalidade, o peso  $w(s, t)$  atribuído para cada arco  $(s, t) \in \mathcal{A}$  é uma combinação linear de um peso baseado na imagem  $0 \leq w_i(s, t) \leq K$  e um peso baseado na pertinência do objeto  $0 \leq w_o(s, t) \leq K$ , que leva em conta todas as  $c$  classes de objetos.

$$w(s, t) = \lambda w_o(s, t) + (1 - \lambda) w_i(s, t) \quad (4.23)$$

onde  $0 \leq \lambda \leq 1$ . Os pesos  $w_i(s, t)$  exploram somente atributos da imagem para capturar descon- tinuidades entre regiões homogêneas. Os pesos  $w_o(s, t)$  levam em conta atributos da imagem para pixels marcados pelo usuário dentro de cada objeto  $l = 1, 2, \dots, c$ . Os marcadores visam extrair infor-

mações de descontinuidade entre o objeto selecionado e o restante da imagem. O parâmetro  $\lambda$  pode ser ajustado e marcadores podem ser adicionados ou removidos para recalculer os pesos dos arcos. As Figuras 4.12a-f ilustram o processo de segmentação de um objeto (vaca) na imagem através da abordagem não-supervisionada seguida da abordagem interativa. Os marcadores da Figura 4.12a geram a distribuição de probabilidade  $\mathcal{P}(s | w_{obj})$  ilustrada na Figura 4.12b. A Figura 4.12c mostra a magnitude do gradiente utilizados pela IFT incremental para correção de falhas, calculada a partir de  $\lambda = 0.5$ . O resultado do Algoritmo 3, com restrição espacial e filtragem de regiões, é mostrado na Figura 4.12d, onde regiões dividem o objeto (vaca) na imagem com rótulo de fundo. Estas regiões são rotuladas como objeto pela IFT somente dentro das regiões que contém os marcadores de objeto, unindo-as, porém limitada às regiões que não possuem estes marcadores, como mostra a Figura 4.12e. Quando uma falha na delimitação ocorre, os pesos obtidos pela Equação 4.23 são utilizados como na IFT de marcadores de objeto (branco) e de fundo (preto) na Figura 4.12f.

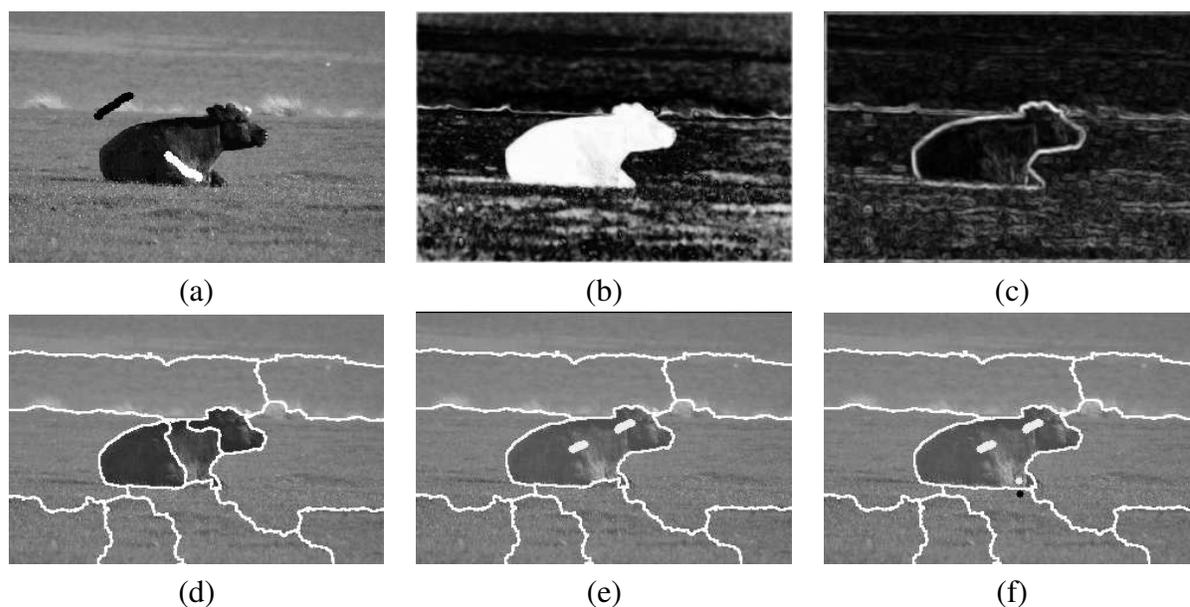


Fig. 4.12: (a) Marcadores de objeto e fundo são utilizados para obter (b) a distribuição de probabilidade condicional do objetos. (c) Pesos para a IFT são obtidos pela combinação linear das magnitudes dos gradientes de brilho e probabilidade condicional do objeto. (d) As partições da imagem obtidas pelo Algoritmo 3 com filtragem e restrição espacial são rotuladas como fundo. (e) Marcadores de objeto unem pela IFT (parcial) as regiões em que estão envolvidos. (f) Os marcadores de objeto (branco) e fundo (preto) são impostos pelo usuário onde ocorrem falhas no delineamento por OPF, que usa a IFT com o gradiente em (c) para correção.

### 4.3.5 Aplicações em Diversos Conjunto de Dados

O próximo Capítulo irá apresentar resultados da aplicação das abordagens apresentadas neste Capítulo, para diversas bases de dados. O agrupamento será utilizado para aplicações de agrupamento de dados multidimensionais obtidos de diversos descritores, como descritores de forma, ou gerados artificialmente para avaliação. Também serão abordadas aplicações em imagens, 2D e 3D, em que se tornam necessárias restrições de domínio ou amostragem no conjunto de dados para tornar mais rápida a execução do algoritmo.

# Capítulo 5

## Análise Comparativa e Resultados Experimentais

O objetivo deste Capítulo é apresentar uma análise comparativa entre métodos de segmentação e resultados e experimentais relacionados com a técnica de agrupamento proposta.

O algoritmo de agrupamento por OPF (Floresta de Caminhos Ótimos) implementa a técnica de *mean-shift* no domínio discreto usando o arcabouço robusto da IFT. Desta forma, algumas falhas presentes no algoritmo original foram endereçadas naturalmente dentro do contexto de morfologia, sem a necessidade de soluções adaptativas e pós-processamentos. Uma redução de grupos é feita através da remoção de domos na PDF e sua associação com classes de objetos é feita pelo usuário através da união de grupos de uma mesma classe.

### 5.1 Análise Comparativa

O algoritmo de OPF encontra os grupos naturais em uma base de dados, mas não garante o número desejado de grupos. Outros métodos de agrupamento garantem o número de grupos desejado, mas quais grupos correspondem a cada classe não pode ser resolvido apenas baseando-se em funções de similaridade e critérios de otimalidade. O número de grupos por classe é até mesmo desconhecido em várias aplicações. Este problema é ilustrado pela avaliação comparativa entre o algoritmo de OPF e vários métodos de agrupamento em várias bases de dados rotuladas. A comparação das técnicas de segmentação de imagens é feita entre as técnicas sem supervisão e aquelas supervisionadas pelo usuário. Entre as técnicas não-supervisionadas são apresentados resultados dos algoritmos de OPF, *mean-shift* e WMT (*watershed* de marcadores rotulados). A comparação entre OPF com correção pela IFT e a segmentação através de *watershed* é feita através do esforço necessário para eliminar grupos irrelevantes e unir aqueles que fazem parte do conjunto de classes desejado pelo usuário.

## 5.2 Comparações Experimentais com Outros Métodos

Considere uma base de dados rotulada  $\mathcal{N}$ , onde a classe de cada amostra é conhecida. Uma boa abordagem de agrupamento deve achar os grupos naturais sem misturar amostras entre classes distintas. Fixando o número de grupos igual ao número de classes, por exemplo, é possível observar misturas entre classes distintas de uma base de dados. Para os experimentos desta seção, foram selecionadas bases de dados sintéticas (Figuras 5.1 e 5.2), nas quais é esperado um grupo por classe, e três bases de dados reais, MPEG-7 [96], Wisconsin Breast Cancer (WBC) [97] e Letter Recognition (LR) [97], nas quais não é possível saber o número de grupos por classe. MPEG-7 consiste de formas (Figura 5.3) que são agrupadas usando três descritores de forma: *Beam Angle Statistics* (BAS) [98], *Fourier Coefficients* (FC) [99], e *Multi-Scale Fractal dimensions* (MSF) [100]. Estes descritores provêm diferentes graus de separação de classe no espaço de características. Um desempenho esperado é um tanto superior como as classes são separadas no espaço de características.

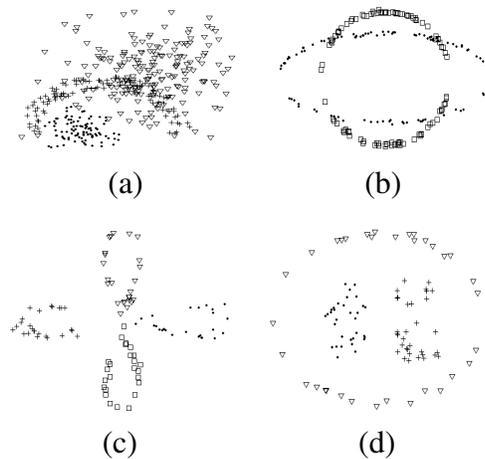


Fig. 5.1: Bases de dados de pontos 2D: (a) cone-torus, (b) saturn, (c) petals, e (d) boat.

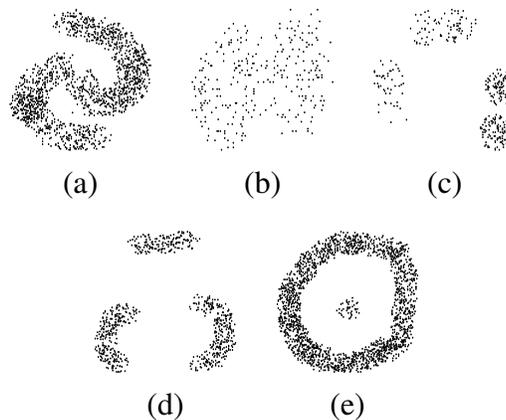


Fig. 5.2: Bases de dados de pontos 2D: (a) data1, (b) data2, (c) data3, (d) data4 e (d) data5.

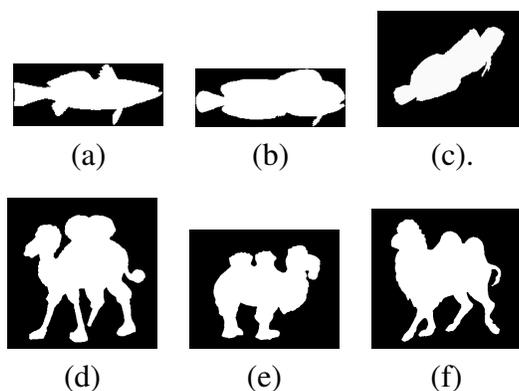


Fig. 5.3: Exemplos de formas MPEG-7 de classes (a)-(c) peixe e (d)-(f) camelo.

Todas as amostras em uma árvore de caminho ótimo dada são assumidas como tendo o mesmo rótulo de sua raiz. Para medir a mistura de classes entre grupos, pode-se verificar as raízes da floresta, designar a classe correta para o rótulo de cada raiz e propagar este rótulo para as amostras restantes de sua árvore de caminhos ótimos. A *pureza* de um agrupamento é então medido como o percentagem de classificações corretas deste procedimento. Para outros métodos que não baseados em árvores de caminhos ótimos, cada grupo é atribuído à classe da maioria de suas amostras e então utilizada a medida de pureza.

Foi escolhida a biblioteca CLUTO <sup>1</sup> para os experimentos porque ela provê seis métodos de agrupamento, quatro funções de similaridade e doze critérios de otimalidade. Foram avaliadas todas as possíveis combinações entre métodos de agrupamento, função de similaridade critérios de otimalidade e um código foi atribuído para cada combinação com valores de pureza mais altos, como mostra a Tabela 5.1. Biblioteca CLUTO é utilizada em seus parâmetros. Os melhores métodos foram: *graph* - computa um agrupamento de  $c$  partições de grafo de vizinhos mais próximos pelo método de corte mínimo; *bagglo* - abordagem aglomerativa em  $c$  grupos; e *rbr* - abordagem particional em  $c$  grupos com otimização global. Para este último caso, cada função de similaridade apresentou o melhor resultado: *cos* - função cosseno; *corr* - correlação de coeficientes; *dist* - inverso da distância Euclidiana; e *jacc* - coeficiente de Jaccard estendido. Os melhores critérios de otimalidade foram: *i2* - maximização da similaridade total de cada grupo; *clink* - critério de *complete-link* tradicional; e *glp* - minimização da similaridade entre grupos distintos.

O número  $c$  de classes da base de dados é utilizadas como o número de grupos resultantes da CLUTO. Valores de pureza acima de 70% podem ser observados na Tabela 5.2 para os casos em que cada classe pode ser representado por um único grupo (data1–data5 e petals, cone-torus, e boat), exceto por saturn. Esta correspondência unívoca não parece ser válida para LR e MPEG-7, mas é mantida para WBC. Os valores de pureza indicam que o descritor BAS pode separar melhor MSF

<sup>1</sup>URL: <http://glaros.dtc.umn.edu/gkhome/views/cluto>

| Código | Método | Similaridade | Critério de otimalidade |
|--------|--------|--------------|-------------------------|
| 1      | graph  | dist         | i2                      |
| 2      | graph  | jacc         | i2                      |
| 3      | bagglo | cos          | clink                   |
| 4      | rbr    | corr         | g1p                     |

Tab. 5.1: Código para as melhores combinações de método, similaridade, função e critério de otimalidade na biblioteca CLUTO. Os valores de pureza para estas combinações em seus respectivas bases de dados são listadas na Tabela 5.2. É usada a mesma nomenclatura da biblioteca CLUTO para seus parâmetros.

e FC. A Tabela 5.2 também apresenta valores de pureza obtidos pelo OPF para cada base de dados. O critério de otimalidade é a floresta de caminhos ótimos com o corte mínimo normalizado (Equação 4.5) e a função de similaridade é equivalente a *dist* na Tabela 5.1. Os resultados do algoritmo de OPF podem ser melhorados com outras funções de similaridade e critérios de otimalidade para um valor de  $k$ -vizinhos. Os parâmetros  $k_{max}$  e  $H$  foram encontrados experimentalmente usando abertura de volume na PDF [101]. A idéia geral é minimizar o número grupos para valores de pureza acima de 70%. Para as bases data1–data5, petals, boat e cone-torus, o algoritmo de OPF bons resultados de pureza, alguns superiores à CLUTO, com o número desejado de classes, com exceção de saturn. Para WBC foram necessários quatro grupos para obter um resultado similar ao da CLUTO. Nas bases MPEG-7, a separabilidade é tanto superior quanto menor é o número de grupos obtidos para cada descritor, mas o número de grupos é ainda maior que o número de classes. Qualquer tentativa de reduzir o número de grupos reduz drasticamente os valores de pureza.

### 5.2.1 Segmentação Não-Supervisionada

O algoritmo de OPF utiliza uma sub-amostragem para reduzir o total de pixels utilizado na estimação do parâmetro de  $k$ -vizinhos em um grafo completo. Uma razão de 16 para 1 é suficiente para imagens 2D, mas razões podem ser aplicadas para imagens 3D, de forma que o tempo de execução que varia de segundos a pouco minutos. Os resultados são mostrados através do nível de cinza das imagens, apesar de terem versões originais coloridas das mesmas imagens. Essas imagens são utilizadas para ilustrar a argumentação de uma análise comparativa entre os algoritmos de *watershed* de marcadores rotulados, *mean-shift* e agrupamento de pixels por OPF.

A técnica de mean-shift revisitada por Comaniciu e Meer tem sua principal aplicação em segmentação de imagens e rastreamento de objetos em vídeo e sua formulação será descrita aqui em termos de sua implementação original pelos autores <sup>2</sup>. A abordagem da OPF têm aplicações mais gerais e depende menos de parâmetros *ad hoc* do usuário.

<sup>2</sup>Código-fonte disponível em <http://www.caip.rutgers.edu/riul/research/code/EDISON>

| Bases de dados (classes) | CLUTO (código) | OPF (grupos) |
|--------------------------|----------------|--------------|
| data1 (2)                | 99.37 (1)      | 99.09 (2)    |
| data2 (2)                | 98.59 (1)      | 97.53 (2)    |
| data3 (5)                | 88.24 (1)      | 99.71 (5)    |
| data4 (3)                | 74.64 (3)      | 100.00 (3)   |
| data5 (2)                | 97.73 (2)      | 100.00 (2)   |
| LR (26)                  | 39.43 (1)      | 70.83 (256)  |
| MPEG7-BAS (70)           | 57.36 (4)      | 82.86 (258)  |
| MPEG7-FC (70)            | 33.36 (1)      | 76.86 (671)  |
| MPEG7-MSF (70)           | 43.29 (1)      | 77.00 (587)  |
| petals (4)               | 100.00 (2)     | 98.00 (4)    |
| saturn (2)               | 58.00 (2)      | 82.50 (13)   |
| boat (3)                 | 79.00 (1)      | 74.00 (3)    |
| cone-torus (3)           | 72.00 (1)      | 72.00 (3)    |
| WBC (2)                  | 95.70 (1)      | 94.84 (4)    |

Tab. 5.2: As colunas mostram bases de dados, valores de pureza obtidas pelos métodos na CLUTO e OPF, com os respectivos número de classes, códigos de combinação e número mínimo de grupos entre parênteses, indicados na Tabela 5.1 e no texto.

O agrupamento por OPF é implementado através do Algoritmo 3 com função de caminho  $f_2$  (Equação 4.11), que calcula a zona de influência de todos os máximos se  $H$  é dado pela Equação 4.10. Com o objetivo de evitar uma super-segmentação, o número de máximos irrelevantes pode ser reduzido com a escolha de  $d_i$  e  $H$ . Figuras 5.4a e 5.4b ilustram o problema, para características de 5 escalas, com  $d_i = 7$  e o valor inicial de caminho  $H$  calculado pela Equação 4.10. Valores menores de  $d_i$  aumentam o problema de super-segmentação. Pela escolha de  $H$  como abertura de volume em  $\rho$ , o número de regiões irrelevantes é consideravelmente reduzido (Figura 5.4c). Resultados semelhantes podem ser obtidos com o algoritmo de *mean-shift* [8] seguido pela união de regiões [12]. Por exemplo, é possível aplicar critério de área mínima para a formação de um grupo, rotular grupos pelo agrupamento de pixels que também estão próximos no espaço de característica, e juntar grupos cujos máximos estão a uma distância que é menor ou igual a  $d_f$  (Figura 5.4d). Nossa abordagem provê um mecanismo adicional de filtragem pela escolha de  $H$  e que poderia tirar vantagem desta união de regiões. Figura 5.5 ilustra outros exemplos obtidos pelo uso de algoritmo de *mean-shift* com união de regiões e a abordagem proposta. Percebe-se que a união de regiões pode remover algumas bordas importantes e o agrupamento por OPF pode reduzir consideravelmente o número de grupos sem unir regiões, o que o torna mais geral que o algoritmo de *mean-shift*.

O algoritmo de *mean-shift* calcula as zonas de influência de máximos através do procedimento que se segue. Para cada amostra  $s \in \mathcal{N}$ , é calculada a direção do gradiente da PDF em direção à variação máxima de densidade ao redor de  $s$  sem que a PDF seja calculada explicitamente. A

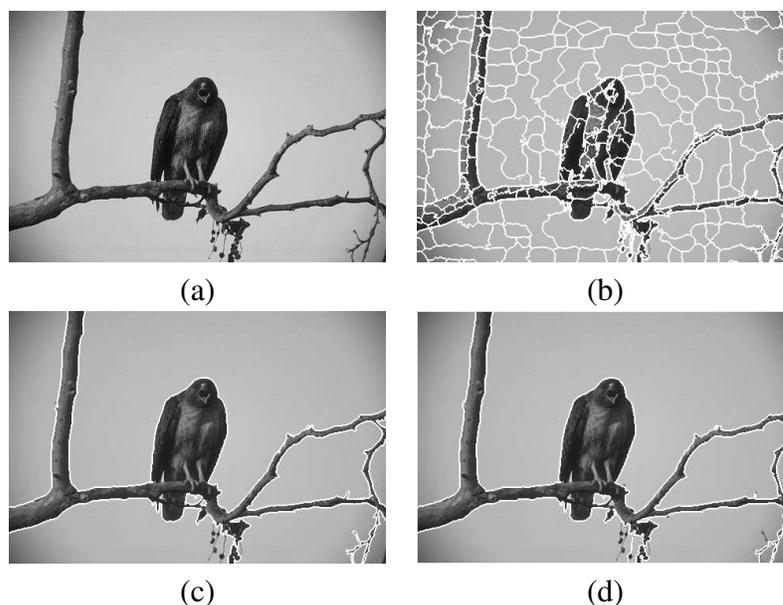


Fig. 5.4: (a) A imagem nível de cinza de um falcão em uma árvore. (b) As bordas dos grupos sobre (a) no caso de uma super-segmentação. (c) O número de regiões irrelevantes é reduzindo pela escolha de  $H$  com base na abertura de volume em  $\rho$ . (d) O resultado do algoritmo de *mean-shift* com a união de regiões.

zona de influência de cada máximo é definida por amostras que o alcançam. O algoritmo calcula o gradiente da PDF iterativamente, através do deslocamento do kernel para a próxima posição. Isto pode apresentar problemas se o vetor de gradiente não é estimado com amostras suficientes ou se o gradiente é zero. Fica claro que é mais robusto definir regiões de influência depois da detecção do máximo, como é feito pelo Algoritmo 3 com  $f_2$ . Neste caso, as regiões onde o gradiente é zero (platôs na PDF) são divididos igualmente entre os máximos que as alcançou, devido ao critério de desempate FIFO em  $Q$ . Por outro lado, se o máximo é representado por pontos vizinhos com mesmo valor de densidade, o algoritmo de *mean-shift* pode repartir a zona de influência deste máximo em múltiplas zonas de influência. Esse problema é naturalmente resolvido com a abordagem da OPF. A Figura 5.6a ilustra um exemplo de uma superfície de uma PDF bidimensional, pela qual caminhos até os dois máximos não apresentam domos ou vales planos como a PDF da Figura 5.6b, onde os maiores valores da PDF correspondem a um platô de um máximo local e os máximos são isolados por vales onde o valor da PDF é zero.

No caso de grafo assimétricos, essa situação é identificada durante a execução do algoritmo e nós em um platô que não possuem arcos simétricos entre si são rotulados como pertencentes a uma mesma zona de influência. a escolha de  $H$  é equivalente ao cálculo de  $V$  de reconstrução morfológica inferior com teFalcao04a. Isto é,  $V$  é a função filtrada que possui máximos eliminados de  $\rho$  por reconstrução. No domínio da imagem, tem sido mostrado que este operador IFT também produz zonas de influência

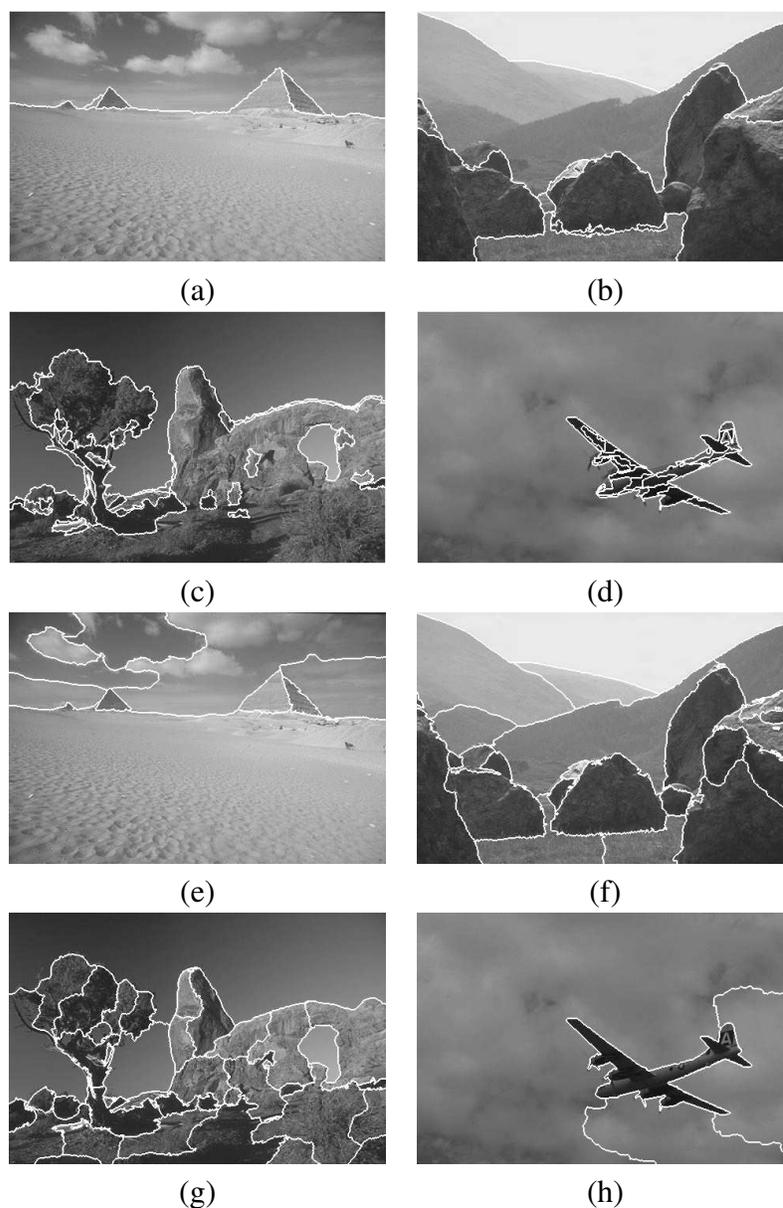


Fig. 5.5: (a-d) Exemplos obtidos pelo algoritmo de *mean-shift* com a união de regiões. (e-h) Resultados com o método proposto para as mesmas imagens.

de todos os máximos da imagem filtrada [20] — isto é, o transformada dual de *watershed* pela IFT a partir marcadores rotulados [2].

A transformada *watershed* pela IFT de marcadores rotulados em níveis de cinza (*IFT-watershed*) pode produzir resultados similares àqueles obtidos pelo Algoritmo 3, usando menos parâmetros mas sendo mais restrito a escolha de  $d_i$ . Figura 5.7a ilustra uma imagem de ressonância magnética de um pulso, contendo ossos e veias. Para prover resultados semelhantes, a transformada *IFT-watershed* de marcadores rotulados utiliza um gradiente multi-escalas estimado pela Equação 4.17. A sua imple-

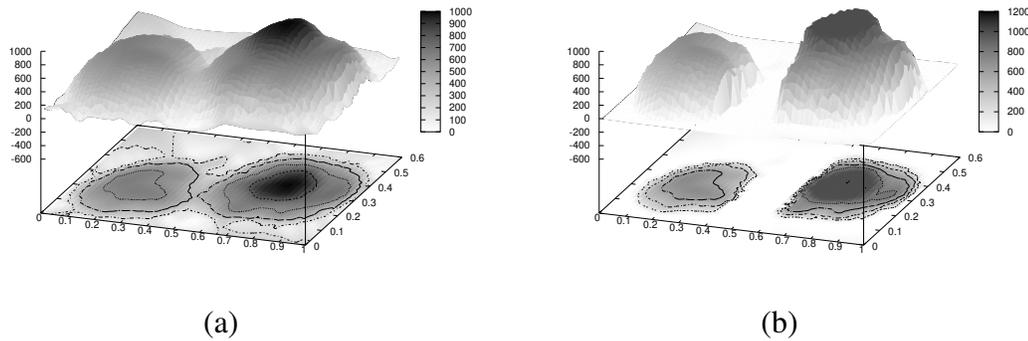


Fig. 5.6: (a) Superfície de uma PDF bidimensional, pela qual caminhos até os dois máximos onde domos e vales não são planos. (b) PDF em que os maiores valores correspondem a um platô de um máximo local e os máximos são isolados por vales onde seu valor é zero.

mentação é obtida pela minimização de valor de caminho  $f_{max}$  (Equação 3.7) para a seguinte relação de adjacência baseada somente no domínio da imagem

$$t \in \mathcal{A}(s) \quad \text{Se } \|t - s\| \leq d_i, \quad (5.1)$$

onde  $d_i$  é a máxima distância Euclidiana entre pixels  $s$  e  $t$  no domínio da imagem.

Apesar dos resultados do algoritmo de *mean-shift* serem bastante similares aos obtidos pelo Algoritmo 3, é preciso ter cautela durante o processo de comparação experimental de implementações diferentes. A implementação do algoritmo de *mean-shift* requer ajustes em parâmetros, uso de características e critério de união de regiões que são diferentes dos utilizados no Algoritmo 3. Por exemplo, um critério baseado em distância entre máximos de cada grupo é utilizado para união de regiões [12]. O mesmo critério poderia ser utilizado na abordagem deste trabalho, no entanto não garantiria que o objeto e o fundo seriam separados. Desta forma, o objetivo do processo do agrupamento é reduzir o número de regiões e deixar a conclusão do processo sob controle do usuário (Seção 4.3.4).

As Figuras 5.8a- 5.8d apresentam grupos rotulados pelo Algoritmo 3 por  $f_2$  com  $H(t) = \rho(t) - 1$  e  $\rho(t) \in [1, 100]$  (Figuras 4.6e- 4.6h). Estes resultados são similares aos gerados pela abordagem do *mean-shift* [8] com a união de zonas de influência de amostras com um mesmo máximo, no entanto, resolve problemas em platôs de densidade. Os objetos são divididos em várias regiões, mas suas bordas são preservadas. De forma a reduzir o número de regiões para segmentação interativa, o Algoritmo 3 é executado com  $H$  calculado por abertura de volume em  $\rho$ [101] Figuras 5.8e- 5.8h).

A transformada IFT-*watershed* de marcadores rotulados em níveis de cinza usa fechamento de volume para criar o marcador  $H(t) > G(t)$  e executa a IFT em um grafo da imagem  $(\mathcal{N}, \mathcal{A}_8)$  para

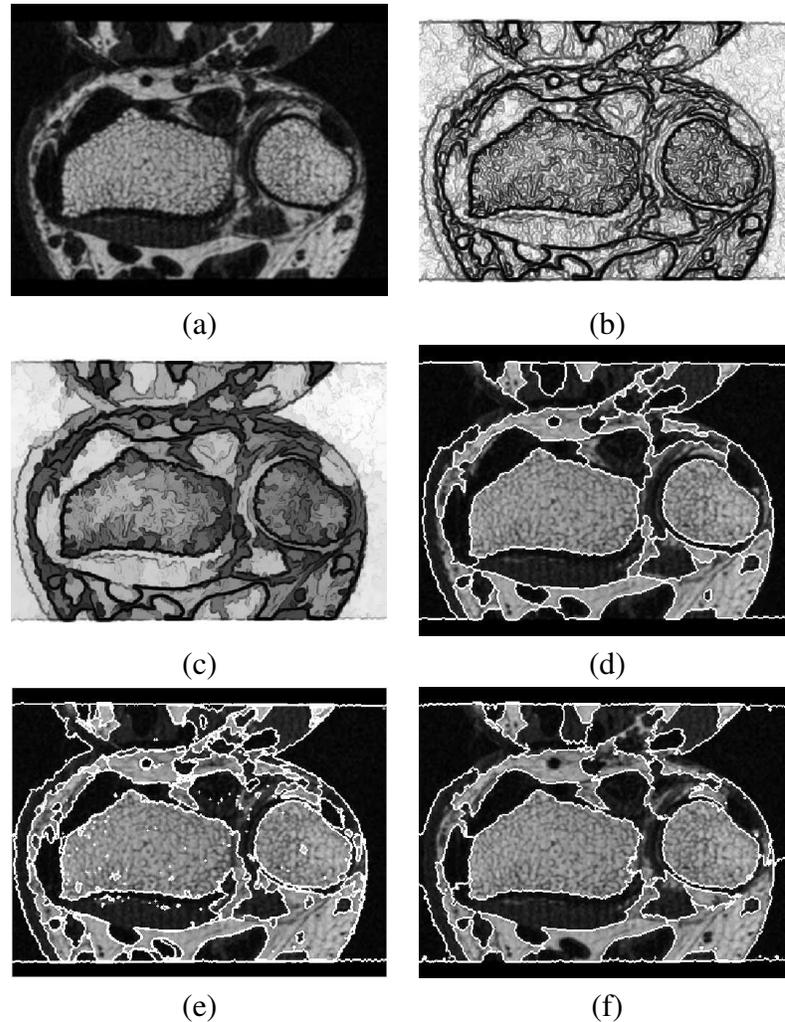


Fig. 5.7: (a) Uma fatia de ressonância de um pulso com ossos e veias. (b) A magnitude do gradiente da Equação 4.17 é usado como valores para os arcos no grafo e (c) os valores iniciais  $h(s)$  para os nós (invertidos e equalizados para ilustração) são entrada da transformada *I níveis de cinza* com  $d_i = 2.8$  e abertura de volume de  $50 \cdot 10^3$  pixels. (d) Os objetos, veias e ossos, são corretamente segmentados com poucos defeitos. (e) O resultado para o Algoritmo 3, com  $d_i = 5$  e abertura de volume de  $15 \cdot 10^3$  pixels, tem resultado com menos artefatos que (f) o resultado do algoritmo de mean-shift, ambos usando apenas a característica de luminância.

minimizar a função de conectividade  $f_{max}$  (Equação 3.7). A restrição por  $d_f$  na adjacência da Equação 4.22 permite um raio maior,  $d_i = 5$  do que o usado na Equação 5.1. Além disto, o uso de  $\rho$  ao invés do gradiente  $G$  reduz em geral o número de regiões do método proposto com respeito ao número obtido pela transformada IFT-*watershed* de marcadores rotulados (Figuras 5.8i- 5.8l).

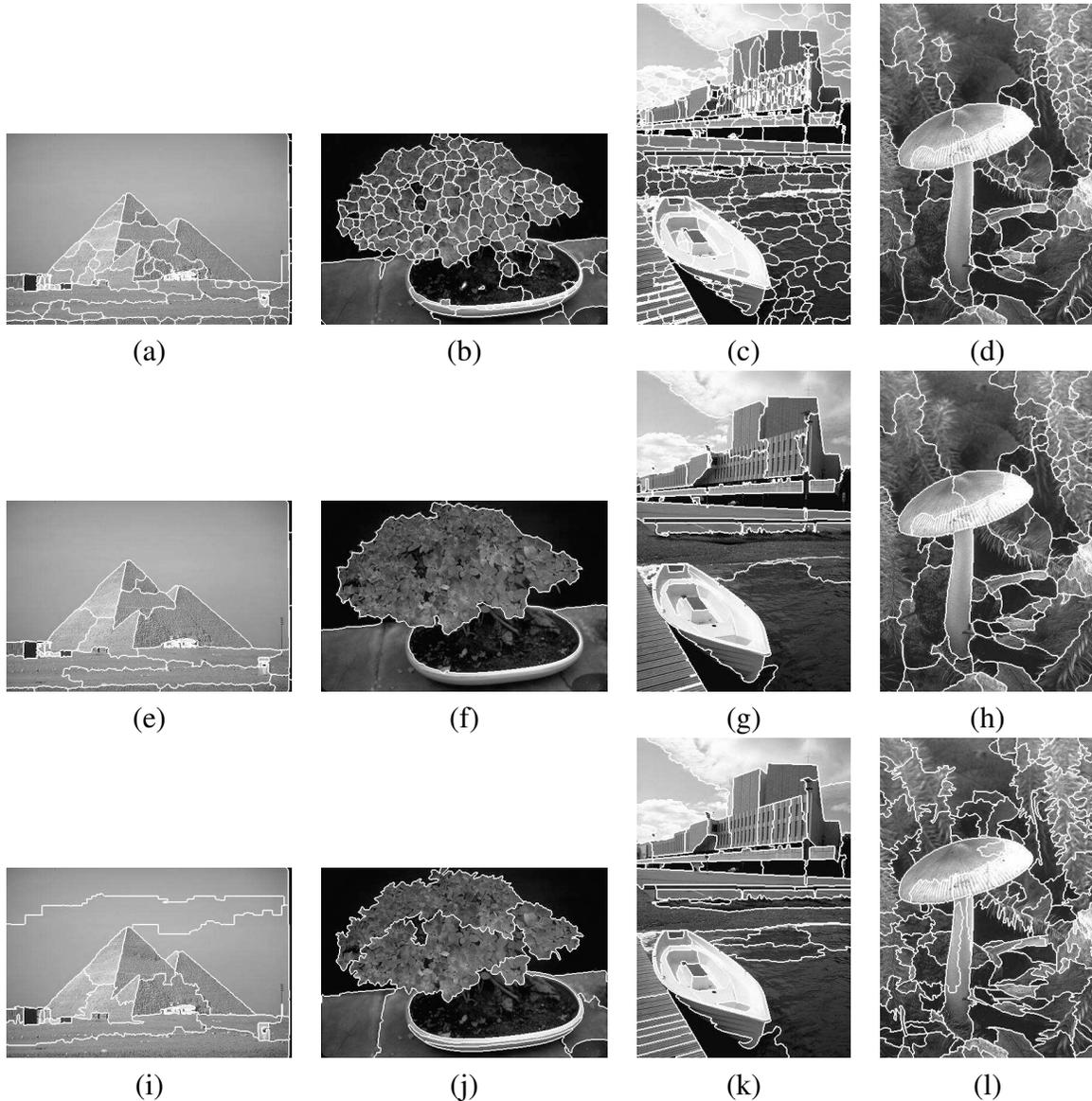


Fig. 5.8: Resultado de agrupamento usando o Algoritmo 3 e  $f_2$  com (a)-(d)  $H(t) = \rho(t) - 1$  e (e)-(h)  $H$  na abertura de volume de  $\rho$ . (i)-(l) Resultados com IFT-*watershed* de marcadores rotulados [2].

### 5.2.2 Segmentação Supervisionada

As regiões na Figuras 5.8e- 5.8h são obtidas pela separação de grupos em componentes 4-conexos. A partição ajuda o usuário a identificar quais regiões compõe o objeto e selecionar marcadores para juntá-las (Figures 5.9a- 5.9d). Isto também mostra quando uma região inclui objeto e fundo (Figura 5.8e), mas seus pixels podem ser facilmente separados com um transformada IFT-*watershed* de marcadores rotulados [50] restritos àquela região. Os marcadores são rotulados como sementes internas e externas, formando o conjunto  $S$ . O Algoritmo de IFT executa em grafo da imagem  $(\mathcal{N}, \mathcal{A}_8)$

pela mesma razão apresentada na Seção 5.2.1. O objeto é definido pela floresta de caminhos ótimos com raízes nas sementes internas.

Figuras 5.9e- 5.9h mostram o resultados da segmentação a partir marcadores e regiões das Figuras 5.9a- 5.9d. Resultados semelhantes poderiam ser obtidos com o gradiente da imagem nas Figuras 4.7a- 4.7d usando apenas a transformada IFT-*watershed* de marcadores rotulados (Figuras 4.7a- 4.7d). De qualquer modo, o método proposto ajuda o usuário a achar diretamente as regiões efetivas para marcadores, geralmente reduzindo o número destes e o também o envolvimento do usuário.

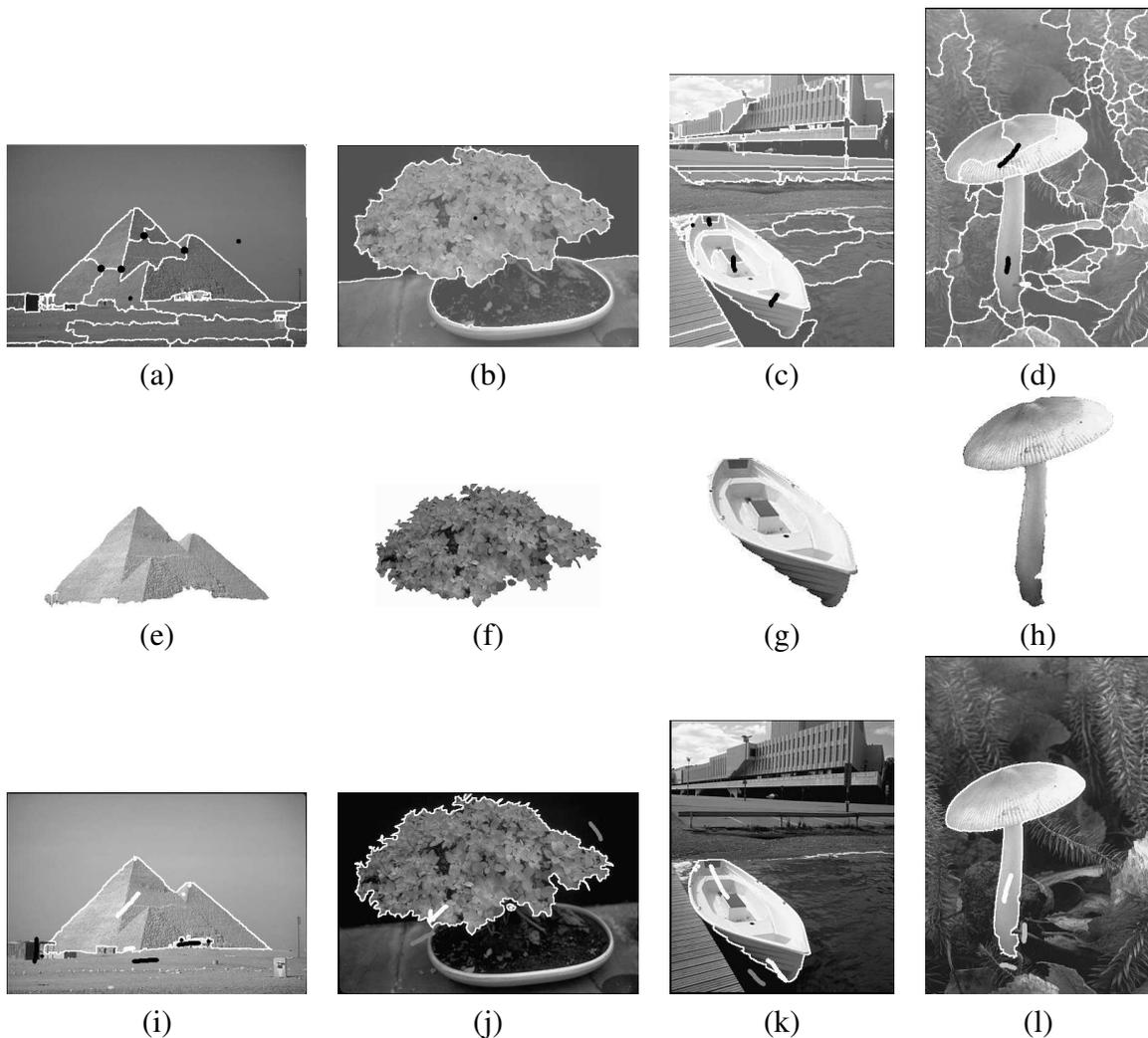


Fig. 5.9: (a)-(d) O usuário seleciona marcadores para juntar regiões e/ou separar dada região. (e)-(h) Resultados das segmentações. (i)-(l) Resultados similares com a transformada IFT-*watershed* de marcadores rotulados. O envolvimento do usuário pode ser diminuído com a guia visual de (a)-(d).

## 5.3 Resultados Experimentais

Nesta seção são apresentados resultados de segmentação de imagens de forma automática e assistida pelo usuário, e classificação automática de grande volume de dados. As aplicações automáticas utilizam técnicas de agrupamento ou classificação não-supervisionada, enquanto a segmentação assistida pelo usuário utiliza uma abordagem interativa.

### 5.3.1 Segmentação Automática de Imagens Naturais

Os resultados apresentados nessa seção utilizam uma base de dados de segmentação de Berkeley [102] (The Berkeley Segmentation Dataset and Benchmark) cujo objetivo é prover uma base empírica para pesquisa em segmentação de imagem e detecção de bordas. Até então foram armazenadas 12.000 segmentações manuais de 30 usuários diferentes a partir de 1.000 imagens da base de dados Corel. Metade das segmentações foram obtidas apresentando ao usuário a imagem colorida; a outra metade foi obtida com a apresentação de imagem em nível de cinza. O teste é baseado no conjunto de segmentações coloridas e nível de cinza para 300 imagens. As imagens são divididas em um conjunto de treinamento de 200 imagens e um conjunto de teste de 100 imagens. É importante ressaltar que esse banco de teste foi feito para algoritmos de detecção de bordas, desde gradientes de cor e textura até complexos algoritmos que utilizam aprendizado de bordas. Apesar de que métodos baseados em regiões não visam extrair bordas, mas rotular regiões, esse banco de teste continua sendo referência até mesmo para esta abordagem.

O experimento para essa base é descrito da seguinte forma. Imagens segmentadas por bordas desenhadas por diferentes usuários constituem o gabarito. O resultado do algoritmo de segmentação para uma imagem da base é usada como entrada do teste. Assume-se que essa saída é uma borda suave com um pixel de largura para as bordas, com valor que varia de zero a um, onde altos valores significam um maior confiança na existência da borda. O algoritmo tenta determinar o quanto essa borda suave se aproxima do gabarito.

Em todo caso é necessário uma limiarização no mapa de bordas para compará-lo às bordas do gabarito, e isto é feito em todos os níveis de cinza. A cada nível são calculados dois valores — precisão e revocação (*precision* e *recall*) — e desta maneira é produzida uma curva de precisão-revocação para o algoritmo e o melhor resultado é escolhido. *Precisão* é a probabilidade de um pixel pertencente à borda gerada pelo algoritmo seja um pixel verdadeiro no gabarito. *Revocação* é a probabilidade que um pixel verdadeiro de borda seja detectado. A medida de precisão avalia o quanto de ruído (artefato) está presente na saída do detector, enquanto a medida de revocação avalia o quanto do gabarito é detectado. A curva de precisão-revocação mostra o custo-benefício entre estas duas quantidades — o compromisso entre erros e falsos positivos — de acordo com o limiar do detector.

Embora a curva de precisão-revoção para um algoritmo seja um descritor de seu desempenho, é ainda necessário extrair um único número para representá-lo. É possível fazê-lo para algoritmos cujas curvas não se interceptam e são grosseiramente paralelas. Quando duas curvas de precisão-revoção não se interceptam, então a curva mais distante da origem domina a outra. A medida desta distância é chamada medida- $F$ , que é a média harmônica entre precisão e revocação. O resultado do algoritmo de segmentação é o valor máximo ao longo da curva de precisão-revoção. Tabelas 5.3, 5.4 e 5.5 mostram respectivamente os resultados de precisão, revocação e a média harmônica de precisão-revoção para o algoritmo 3 na base de treinamento de 200 imagens coloridas. Características multi-escala são utilizadas em cada canal, vermelho, azul e verde, para extração de um vetor de  $\vec{v}$  com 15 características, sendo 5 as escalas em cada canal de cor. Os parâmetros de amostragem são fixados em 16 para 1 pixels, verticalmente e horizontalmente, para a determinação do melhor  $k$  com a escala de  $k_{max}$  em 50, 100, 150 e 200 vizinhos mais próximos. A função  $H$  é selecionada entre quatro filtragens em ordem crescente no seu potencial de remoção de máximos irrelevantes, com valores  $5 \cdot 10^3$ ,  $10 \cdot 10^3$ ,  $15 \cdot 10^3$  e  $20 \cdot 10^3$ .

| $k_{max}$ | Filtragem ( $10^3$ pixels) |        |        |        |
|-----------|----------------------------|--------|--------|--------|
|           | 5                          | 10     | 15     | 20     |
| 50        | 0,5233                     | 0,5631 | 0,5890 | 0,6070 |
| 100       | 0,5546                     | 0,5947 | 0,6209 | 0,6391 |
| 150       | 0,5801                     | 0,6247 | 0,6516 | 0,6687 |
| 200       | 0,5954                     | 0,6413 | 0,6664 | 0,6829 |

Tab. 5.3: Medidas de precisão para o conjunto de treinamento de 200 imagens a partir de da variação dos parâmetros  $k_{max}$  e da quantidade de pixels utilizados na filtragem de volume em  $H$ .

| $k_{max}$ | Filtragem ( $10^3$ pixels) |        |        |        |
|-----------|----------------------------|--------|--------|--------|
|           | 5                          | 10     | 15     | 20     |
| 50        | 0,7083                     | 0,6650 | 0,6385 | 0,6174 |
| 100       | 0,6700                     | 0,6239 | 0,5938 | 0,5725 |
| 150       | 0,6425                     | 0,5945 | 0,5611 | 0,5397 |
| 200       | 0,6312                     | 0,5823 | 0,5488 | 0,5266 |

Tab. 5.4: Medidas de revocação do conjunto de treinamento de 200 imagens a partir de da variação dos parâmetros  $k_{max}$  e  $d$  na filtragem de volume em  $H$ .

A Tabela 5.3 e a Figura 5.10a mostram que os valores de precisão aumentam com filtragem de  $H$  com  $5 \cdot 10^3$  pixels e com a procura de  $k$  na maior escala, pois falsas bordas são eliminadas. O contrário é mostrado na Tabela 5.4 e Figura 5.10b em que de valores de revocação decrescentes com

| $k_{max}$ | Filtragem ( $10^3$ pixels) |        |        |        |
|-----------|----------------------------|--------|--------|--------|
|           | 5                          | 10     | 15     | 20     |
| 50        | 0,6019                     | 0,6098 | 0,6128 | 0,6121 |
| 100       | 0,6069                     | 0,6089 | 0,6070 | 0,6040 |
| 150       | 0,6097                     | 0,6093 | 0,6030 | 0,5973 |
| 200       | 0,6127                     | 0,6104 | 0,6019 | 0,5947 |

Tab. 5.5: Médias harmônicas do conjunto de treinamento de 200 imagens a partir de da variação dos parâmetros  $k_{max}$  e da quantidade de pixels utilizados na filtragem de volume em  $H$ .

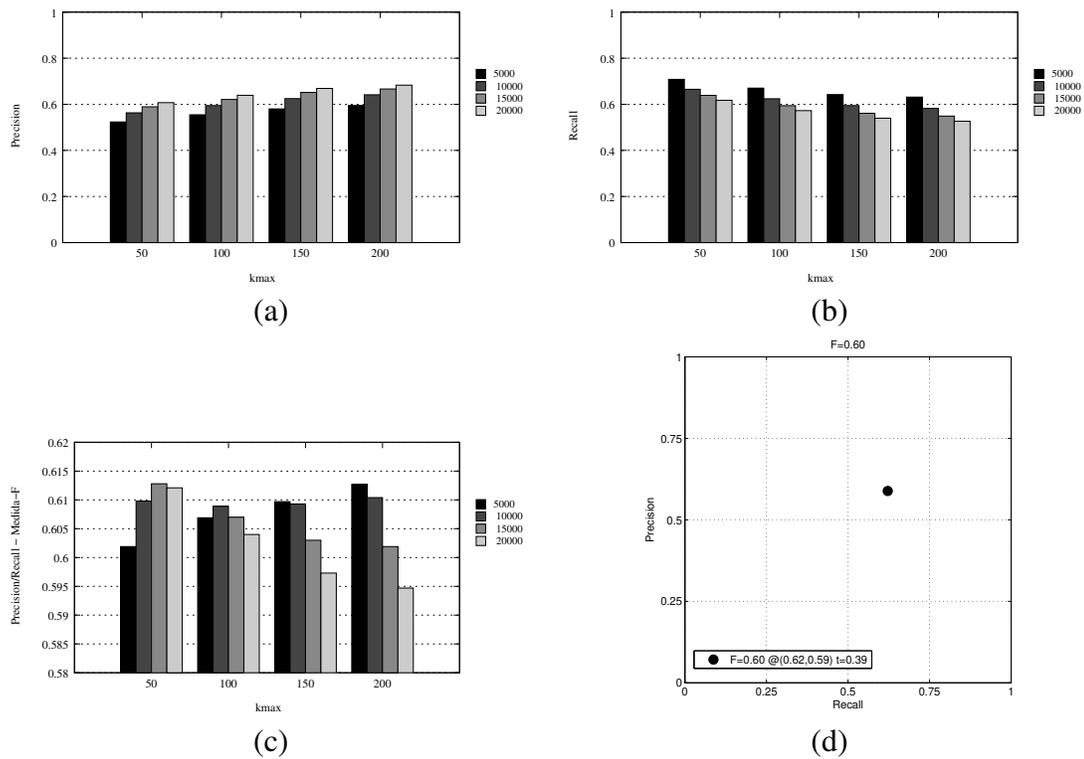


Fig. 5.10: Gráficos dos valores presentes na Tabelas 5.3, 5.4 e 5.5 das respectivas medidas de (a) precisão, (b) revocação e sua (c) média harmônica para o conjunto de treinamento. (d) Média harmônica resultante no conjunto de teste nos eixos de *precision* e revocação

o aumento da filtragem de  $H$ , pois bordas verdadeiras também são removidas. A média harmônica de precisão-revocação procura balancear essas duas medidas.

Os parâmetros de escala  $k_{max}$  e filtragem com melhor média harmônica de precisão-revocação (Tabela 5.5) são respectivamente os valores de  $k_{max} = 50$  e filtragem de  $15 \cdot 10^3$  pixels, com média harmônica igual a 0.612760. Esse resultado não é muito distante dos valores obtidos em todas escalas e filtrações, com média 0.6064 e desvio padrão 0.0055. Isto mostra que na tarefa de identificar bordas, o algoritmo proposto obtém um bom resultado, mesmo sendo baseado em rotulamento de regiões.

Para a realização do experimento no banco de teste de 100 imagens, os parâmetros com maior valor de precisão-revocação foram utilizados. Os resultados do teste foram os valores de precisão, revocação e média harmônica de 0.5887, 0.6217 e 0.60471, respectivamente (Figura 5.10d). Este último resultado de *precision-recall* está entre os oito melhores do banco de dados de teste, com valor superior à medida de um banco de gradiente de cor, mesmo sendo esta uma abordagem baseada em regiões. A abordagem baseada em bordas de [71] possui resultados na banco de imagens de Berkeley com valores semelhantes aos obtidos pelo Algoritmo 3.

### 5.3.2 Classificação de Estruturas do Cérebro

Seja um par  $(\mathcal{N}, I)$  uma imagem de ressonância do cérebro, onde  $\mathcal{N}$  é o conjunto de voxel e  $I(t)$  a intensidade do voxel. Uma aplicação comum consiste em achar dois grupos de estrutura: massa cinzenta (*Gray Matter — GM*) e massa branca (*White Matter — WM*). O subconjunto  $(\mathcal{N}', \mathcal{A})$  é criado pela amostragem de 0.02% dos voxels de  $\mathcal{N}$ , de forma que metade dos voxels tem valor abaixo do brilho médio dentro do cérebro e a outra metade tem valores acima desse valor. Isto permite um número equilibrado de amostras das duas estruturas — GM e WM. O vetor de característica  $\vec{v}(t)$  consiste de valores de brilho  $I(t)$  do voxel  $t$  e os seus vizinhos-6 no domínio da imagem, desde que dentro do cérebro, o valor do brilho do voxel central é repetido. O melhor valor de  $k$  é encontrado dentro do intervalo  $[1, k_{\max}]$ , com  $k_{\max} = 50$  devido ao problema de escala pela não-homogeneidade e volume parcial. O método geralmente acha 2 grupos com essa escala, mas é possível que sejam encontrados mais de dois grupos. Neste caso é feita um nova rotulação (abstração de dados) para forçar apenas dois grupos. Este procedimento atribui um rótulo de GM a todos que possuem brilho abaixo da intensidade média do cérebro, e um rótulo de WM caso contrário. A Equação 2.19 é utilizada para classificar os voxels restantes em  $t \in \mathcal{N} \setminus \mathcal{N}'$ .

Um primeiro experimento ( $OPF_1$ ) consistiu em selecionar 8 imagens de cérebros fantasmas (*phantoms*) da base de dados Brainweb<sup>3</sup>, com valores de ruído de 3%, 5%, 7%, e 9%, e não-homogeneidade de 20% e 40%, respectivamente. A imagem gabarito é disponível e a similaridade entre o resultado e o gabarito é calculada pela métrica *Dice*. Cada teste é executado 10 vezes para cada cérebro usando conjuntos de amostras  $\mathcal{N}'$  diferentes e calculando a média e desvio padrão com as similaridades de *Dice*. O mesmo experimento também foi conduzido para 8 imagens reais (com resolução de intensidade de 9 bits) do banco de dados IBSR<sup>4</sup>.

Dois outros experimentos foram realizados,  $OPF_2$ , que utilizou um procedimento de redução da não-homogeneidade [103] para melhorar a acurácia da classificação de um mesmo tecido do cérebro; e  $OPF_2 + Bayes$ , em que a classificação dos voxels restantes é obtida pelo classificador Bayesiano

<sup>3</sup>URL: <http://www.bic.mni.mcgill.ca/brainweb>

<sup>4</sup>URL: [www.cma.mgh.harvard.edu/ibsr](http://www.cma.mgh.harvard.edu/ibsr)

em  $\mathcal{N} \setminus \mathcal{N}'$  a partir do conjunto de treinamento obtido em  $OPF_2$ . Nestes experimentos os voxels do conjunto de treinamento são amostrados em um número ímpar de vezes, por exemplo 7, e o rótulo final é decidido por votação da maioria.

As Tabelas 5.6 e 5.7 mostram as similaridades de *Dice* para as imagens dos cérebros sintéticos (fantasmas) com pares diferentes de ruído e não-homogeneidade, e as Tabelas 5.8 e 5.9 para cérebros reais. Os resultados são similares aos obtidos por abordagens recentes [3], em que similaridades de *Dice* variam em [93%, 95%] para WM e [89%, 92%] para GM, em cérebros sintéticos, e médias de 80% para GM e 88% para WM em cérebros reais. A classificação de GM é bem mais relevante em estudos médicos do que WM e essa diferença na acurácia é desejável na prática. Os resultados obtidos em  $OPF_2$  são superiores aos obtidos por  $OPF_1$  e  $OPF_2 + Bayes$ . A redução de não-homogeneidade e a votação pela maioria melhora o resultado de agrupamento, o que mostra que a conectividade no espaço de características tem importância na classificação.

Também foi mensurado o tempo computacional do aprendizado não-supervisionado (agrupamento por OPF e estimação do melhor valor de  $k$ -vizinhos) e classificação. O execução mais demorada de  $OPF_1$  foi de 84 segundos em um PC Pentium IV 3GHz. Dado que o método usado para essa comparação [3] utilizou um computador similar para seus experimentos, método por OPF é cerca de 30 vezes mais rápido que essa abordagem.

| Cérebro<br>GM | Similaridade <i>Dice</i> média $\pm$ desvio padrão.(%) |                  |                  | Similaridade <i>Dice</i> |
|---------------|--|------------------|------------------|--------------------------|
|               | $OPF_1$  | $OPF_2$          | $OPF_2 + Bayes$  | <i>Awate</i> [3]         |
| 1 (3%,20%)    | 95.15 $\pm$ 0.17                                       | 95.47 $\pm$ 0.05 | 95.50 $\pm$ 0.02 | 91.32                    |
| 2 (5%,20%)    | 95.10 $\pm$ 0.17                                       | 95.30 $\pm$ 0.05 | 95.51 $\pm$ 0.04 | 90.78                    |
| 3 (7%,20%)    | 94.36 $\pm$ 1.03                                       | 95.49 $\pm$ 0.02 | 95.00 $\pm$ 0.08 | 90.13                    |
| 4 (9%,20%)    | 94.06 $\pm$ 0.27                                       | 94.95 $\pm$ 0.01 | 93.98 $\pm$ 0.04 | 89.32                    |
| 5 (3%,40%)    | 90.90 $\pm$ 1.28                                       | 93.57 $\pm$ 0.07 | 93.50 $\pm$ 0.03 | 91.32                    |
| 6 (5%,40%)    | 91.23 $\pm$ 1.25                                       | 93.27 $\pm$ 0.08 | 93.51 $\pm$ 0.04 | 90.78                    |
| 7 (7%,40%)    | 91.10 $\pm$ 0.72                                       | 93.50 $\pm$ 0.03 | 92.91 $\pm$ 0.05 | 90.13                    |
| 8 (9%,40%)    | 90.66 $\pm$ 1.21                                       | 92.84 $\pm$ 0.02 | 92.30 $\pm$ 0.04 | 89.32                    |

Tab. 5.6: Classificação de GM de cérebros sintéticos (fantasmas) com pares de ruído e não-homogeneidade entre parênteses: média e desvio padrão de similaridades *Dice* em  $OPF_1$ ,  $OPF_2$ ,  $OPF_2 + Bayes$  e na abordagem de Awate et.al. [3].

A não-homogeneidade parece ser o maior desafio, pois não é difícil encontrar regiões diferentes do cérebro onde GM e WM têm intensidades similares (Figura 5.11a). Os erros são concentrados na borda entre GM e WM, onde a variação devido a não-homogeneidade é sensivelmente maior que a variação de ruído (Figura 5.11b).

É interessante salientar que o método de classificação baseado em OPF não melhora significativamente a acurácia pelo incremento da taxa de amostragem de 0.02% até 0.12%. Trabalhos futuros

| Cérebro<br>WM | Similaridade <i>Dice</i> média $\pm$ desvio padrão.(%) |                  |                  | Similaridade <i>Dice</i><br><i>Awate</i> [3] |
|---------------|--|------------------|------------------|--|
|               | $OPF_1$  | $OPF_2$          | $OPF_2 + Bayes$  |  |
| 1 (3%,20%)    | 93.43 $\pm$ 0.19                                       | 94.10 $\pm$ 0.04 | 93.74 $\pm$ 0.06 | 94.85  |
| 2 (5%,20%)    | 93.40 $\pm$ 0.20                                       | 93.89 $\pm$ 0.04 | 93.75 $\pm$ 0.09 | 94.27  |
| 3 (7%,20%)    | 92.55 $\pm$ 0.93                                       | 93.91 $\pm$ 0.02 | 92.79 $\pm$ 0.16 | 93.66  |
| 4 (9%,20%)    | 91.93 $\pm$ 0.54                                       | 93.08 $\pm$ 0.05 | 91.01 $\pm$ 0.09 | 92.94  |
| 5 (3%,40%)    | 88.30 $\pm$ 0.64                                       | 91.75 $\pm$ 0.06 | 91.23 $\pm$ 0.04 | 94.85  |
| 6 (5%,40%)    | 88.19 $\pm$ 0.67                                       | 91.40 $\pm$ 0.05 | 91.04 $\pm$ 0.10 | 94.27  |
| 7 (7%,40%)    | 87.77 $\pm$ 0.81                                       | 91.39 $\pm$ 0.03 | 89.93 $\pm$ 0.13 | 93.66  |
| 8 (9%,40%)    | 87.03 $\pm$ 0.73                                       | 90.45 $\pm$ 0.04 | 88.48 $\pm$ 0.10 | 92.94  |

Tab. 5.7: Classificação de WM de cérebros sintéticos (fantasmas): média e desvio padrão de similaridades *Dice* em  $OPF_1$ ,  $OPF_2$ ,  $OPF_2 + Bayes$  e na abordagem de Awate et.al. [3].

| Cérebro<br>GM | Similaridade <i>Dice</i> média $\pm$ desvio padrão.(%) |                  |                  | Similaridade <i>Dice</i><br><i>Awate</i> [3] |
|---------------|--|------------------|------------------|--|
|               | $OPF_1$  | $OPF_2$          | $OPF_2 + Bayes$  |  |
| 1 (3%,20%)    | 93.43 $\pm$ 0.19                                       | 94.10 $\pm$ 0.04 | 93.74 $\pm$ 0.06 | 94.85  |
| 2 (5%,20%)    | 93.40 $\pm$ 0.20                                       | 93.89 $\pm$ 0.04 | 93.75 $\pm$ 0.09 | 94.27  |
| 3 (7%,20%)    | 92.55 $\pm$ 0.93                                       | 93.91 $\pm$ 0.02 | 92.79 $\pm$ 0.16 | 93.66  |
| 4 (9%,20%)    | 91.93 $\pm$ 0.54                                       | 93.08 $\pm$ 0.05 | 91.01 $\pm$ 0.09 | 92.94  |
| 5 (3%,40%)    | 88.30 $\pm$ 0.64                                       | 91.75 $\pm$ 0.06 | 91.23 $\pm$ 0.04 | 94.85  |
| 6 (5%,40%)    | 88.19 $\pm$ 0.67                                       | 91.40 $\pm$ 0.05 | 91.04 $\pm$ 0.10 | 94.27  |
| 7 (7%,40%)    | 87.77 $\pm$ 0.81                                       | 91.39 $\pm$ 0.03 | 89.93 $\pm$ 0.13 | 93.66  |
| 8 (9%,40%)    | 87.03 $\pm$ 0.73                                       | 90.45 $\pm$ 0.04 | 88.48 $\pm$ 0.10 | 92.94  |

Tab. 5.8: Classificação de GM de cérebros reais (ISBR): média e desvio padrão de similaridades *Dice* em  $OPF_1$ ,  $OPF_2$ ,  $OPF_2 + Bayes$  e na abordagem de Awate et.al. [3].

| Cérebro<br>WM | Similaridade <i>Dice</i> média $\pm$ desvio padrão.(%) |                             |                  | Similaridade <i>Dice</i><br><i>Awate</i> [3] |
|---------------|--|-----------------------------|------------------|--|
|               | $OPF_1$  | $OPF_2$                     | $OPF_2 + Bayes$  |  |
| 1             | 92.22 $\pm$ 0.87                                       | 90.33 $\pm$ 0.09            | 90.34 $\pm$ 0.12 | 83.33  |
| 2             | 90.99 $\pm$ 2.93                                       | 91.72 $\pm$ 0.02            | 87.54 $\pm$ 0.30 | 85.34  |
| 3             | 93.86 $\pm$ 0.14                                       | 91.99 $\pm$ 0.10 $\pm$ 0.13 | 87.25            |  |
| 4             | 88.19 $\pm$ 5.97                                       | 92.32 $\pm$ 0.10            | 90.33 $\pm$ 0.18 | 83.24  |
| 5             | 90.20 $\pm$ 1.73                                       | 90.33 $\pm$ 0.02            | 88.00 $\pm$ 0.09 | 86.41  |
| 6             | 85.02 $\pm$ 4.21                                       | 89.42 $\pm$ 0.05            | 89.68 $\pm$ 0.11 | 81.62  |
| 7             | 91.22 $\pm$ 3.35                                       | 91.34 $\pm$ 0.08            | 87.29 $\pm$ 0.15 | 81.07  |
| 8             | 88.46 $\pm$ 4.39                                       | 90.80 $\pm$ 0.02            | 88.27 $\pm$ 0.10 | 78.06  |

Tab. 5.9: Classificação de WM de cérebros reais (ISBR): média e desvio padrão de similaridades *Dice* em  $OPF_1$ ,  $OPF_2$ ,  $OPF_2 + Bayes$  e na abordagem de Awate et.al. [3].

tentarão endereçar o problema de não-homogeneidade.

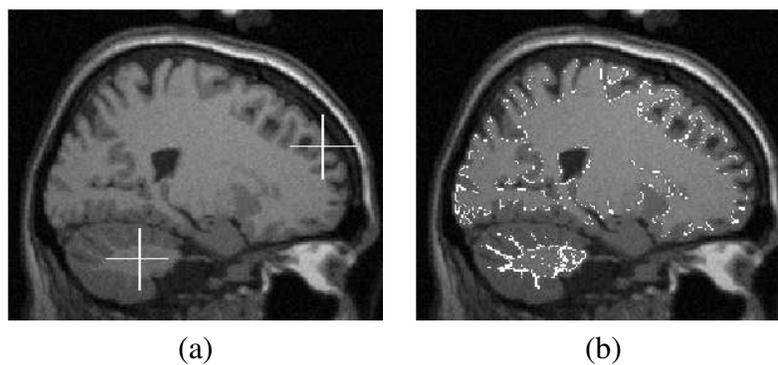


Fig. 5.11: (a) Os marcadores mostram voxels de WM e GM voxels com mesmo valor de intensidade 1642. (b) Os erros (destacados em branco) são concentrados nas bordas entre GM e WM.

# Capítulo 6

## Conclusão

O trabalho desta tese de doutorado apresenta uma metodologia de agrupamento através do arcabouço da Transformada Imagem-Floresta (IFT). Os objetivos alcançados vão além do âmbito teórico do algoritmo de agrupamento e da analogia com *watershed*. Uma nova abordagem é desenvolvida para resolver o problema de segmentação de imagens através da IFT através da sua extensão ao espaço multidimensional de características.

O algoritmo de Floresta de Caminhos Ótimos (OPF) é apresentado como uma elaboração no contexto de matemática discreta através do uso de grafos a partir do algoritmo de *mean-shift*. O uso de grafos apresenta-se como uma abordagem natural ao conjunto de dados, que por definição é finito e discreto. A teoria do procedimento de *mean-shift* basea-se no cálculo implícito da PDF através de seu gradiente. Tal abordagem apresenta bons resultados quando não existem problemas na estimação do gradiente da PDF, como variações bruscas devido a pequena quantidade de amostras dentro da janela de estimação, ou situações em que este é nulo devido a acidentes na topologia da PDF, como platôs de densidade.

A PDF é estimada através de uma janela de Parzen, onde a distância Euclidiana entre características dentro da largura de banda da janela contribuem para o cálculo de probabilidade. Através de características multi-escalas obtidas através de filtragem morfológicas, cada nó do grafo ou sub-grafo possui informação em diversas escalas sem a suavização de bordas produzida por filtragens Gaussianas. O vetor de característica utiliza as bandas de cores da imagem, no caso de imagens coloridas e o brilho da imagem em imagens nível de cinza. No caso de imagens médicas, ao invés da característica multi-escala baseada em filtragens, é utilizada uma característica baseada na probabilidade associada ao brilho do pixel e seus vizinhos. O uso dessa característica é mais apropriada em imagens de ressonância do cérebro, onde a não-homogeneidade gera variações de brilho ao longo de diferentes regiões do tecido de massas cinzenta e branca.

Em um contexto geral, agrupamento por OPF pode ser aplicado em qualquer tipo de dado, com e

sem restrição de domínio. No caso de imagens, duas alternativas são apresentadas devido ao grande número de amostras. A primeira solução é a redução de número de nós permite que o algoritmo não-supervisionado de agrupamento seja executado apenas no sub-grafo gerado e depois uma classificação usando os rótulos do passo anterior rotula as amostras restantes. A outra alternativa é restringir o domínio através de uma adjacência que leva em conta uma correlação dos  $k$ -vizinhos mais próximos. No caso de imagens, a correlação espacial é imposta aos pixels vizinhos ou uma sub-amostragem é feita para criar o sub-grafo que contém os objetos na imagem.

Testes realizados no banco de imagens de Berkeley e comparações com os algoritmos de *watershed* e *mean-shift* mostram que OPF oferece uma flexibilidade e robustez através do uso da filtragem de PDF e do arcabouço da IFT, respectivamente. O resultado obtido pela medida harmônica entre as probabilidades de precisão e revocação está entre melhores algoritmos de segmentação publicados pelo banco de imagens. É importante ressaltar que o gabarito da base de imagens é baseada em bordas, o que impacta no desempenho de uma abordagem baseada em regiões. No entanto, o problema de segmentação de imagens ainda está longe de ser alcançado devido a própria natureza subjetiva do problema, onde o objeto a ser delineado depende da intenção do usuário. Nesse contexto, assim como *watershed*, o algoritmo de OPF pode receber marcadores para achar os máximos regionais com rótulos que são definidos pelo usuário. Do ponto de vista deste, o processo pode não ser intuitivo ou pode exigir uma intervenção excessiva do usuário, através da imposição de muitos marcadores. Uma solução é sugerir ao usuário as regiões da imagem que possuem grande similaridade e que são divididas por regiões. Estas podem ser selecionadas e unidas com marcadores, e sua borda corrigida através do algoritmo de *watershed* caso haja necessidade. Este último apresenta melhores resultados quando o gradiente da imagem é obtido pela combinação do gradiente dos mapas de objeto e fundo da imagem, obtidos por marcadores em um pré-processamento, e um gradiente de textura multi-escala, também apresentado neste trabalho.

Em casos em que o conjunto de dados é muito grande para o cálculo do algoritmo da OPF e é inviável na prática. Seu alto custo computacional se deve à adjacência de  $k$ -vizinhos mais próximos, o cálculo da PDF e a estimação da largura de banda  $h_f$ , cujo valor de  $k$  apresenta o melhor corte no grafo. Assim, um sub-grafo é selecionado para a estimação do melhor valor de  $k$  e sem a restrição espacial e a PDF apresenta um número menor de máximos irrelevantes. A conectividade no espaço de características mostra-se importante, e os rótulos das raízes são utilizados para classificar o restante dos pixels, cujos valores caminhos dados por  $f_{min}$  são maximizados desde cada raiz. O número de grupos gerados no conjunto sub-amostrado é o mesmo no resultado final de agrupamento, o que torna essa técnica robusta para grande bases de dados. Resultados em imagens 3D apresentam resultados similares aos de técnicas no estado-da-arte para classificação de massas cinzenta (GM) e branca (WM) do cérebro. A comparação com o classificador Bayesiano mostra resultados de acurácia similares para

GW e superiores para WM.

## 6.1 Trabalhos Correlatos e Futuros

Trabalhos de mestrado e doutorado em conclusão e em andamento possuem uma relação direta ou indireta com esta tese. Estes trabalhos foram desenvolvidos em conjunto com o Prof. Falcão do Instituto de Computação da Universidade Estadual de Campinas e utilizam a teoria de caminhos ótimos em grafos da IFT para a criação de classificadores e outras aplicações baseados em grafos [104, 105].

No trabalho de doutorado [72, 106], é proposto um classificador supervisionado que utiliza um grafo completo ponderado somente nos arcos. Amostras de classes distintas do conjunto de dados de treinamento são utilizadas para criar nós de suporte do classificador. Árvores de caminhos ótimos são criadas através da minimização da função de caminho  $f_{max}$ , como no algoritmo de *watershed* descrito na Seção 3.3.3. Em contraste com o agrupamento por OPF, os nós de suporte ficam concentrados na periferia das classes, ao invés dos máximos na PDF, geralmente concentrados nos centros das classes.

O trabalho de mestrado [107] desenvolveu uma técnica de recuperação de informações baseada no algoritmo de OPF. Em cada busca realizada em um banco de imagens uma imagem é utilizada como entrada. O comportamento de um usuário é simulado através da divisão das imagens retornadas em dois grupos: “relevantes” e “irrelevantes”. O procedimento de busca baseia-se em agrupamento por OPF, onde as imagens são nós do grafo e os arcos são definidos pela relação de adjacência dos  $k$ -vizinhos mais próximos. A metodologia é a mesma da anteriormente apresentada no Capítulo 4, com características e métricas apropriadamente selecionadas para realização da tarefa. A cada iteração, as imagens classificadas pelo usuário podem ser utilizadas para inferir quais os critérios que o usuário está utilizando. A técnica proposta atribui uma pontuação para cada grupo levando em conta a porcentagem de imagens relevantes e irrelevantes no agrupamento. São priorizadas as imagens dos grupos que tiverem a melhor pontuação, retornando-as na próxima busca.

A análise de assimetrias dos ventrículos laterais do cérebro foi motivação do trabalho de mestrado [108], onde é feita uma análise de assimetria dos ventrículos laterais. Através da segmentação dessas estruturas é possível auxiliar o profissional médico no diagnóstico de doenças cerebrais relacionadas a alterações no volume e na forma. Um cérebro saudável apresenta um alto grau de simetria relativa ao plano sagital e que uma assimetria relativa a este plano pode ser um indicador de doenças como epilepsia, mal de Alzheimer e de existência de tumores cerebrais. Assimetrias volumétricas, adicionalmente, podem estar associadas a doenças como esquizofrenia, depressão e demência. A tarefa de segmentação dos ventrículos laterais depende de bons resultados na segmentação da superfície do cérebro, que é envolta pelo líquido cefalorraquidiano — CSF. O trabalho ainda em desenvolvi-

mento utiliza um algoritmo baseado no OPF para remover o CSF e analisar os grupos de pacientes com assimetria em relação ao grau de assimetria de forma dos ventrículos laterais do cérebro em relação ao grupo controle. Com os avanços e melhorias na segmentação do CSF, é de se esperar que a segmentação dos ventrículos laterais sofra um aprimoramento expressivo.

Trabalhos desenvolvidos em conjunto [109, 103] com o doutorado em andamento no Instituto de Computação utiliza a técnica de agrupamento por OPF para separar as substâncias cinzenta e branca do cérebro. Os resultados presentes na Seção 5.3.2 fazem parte de uma pesquisa conjunta durante o desenvolvimento da técnica. Futuros resultados poderão ser obtidos com a incorporação de outras técnicas de segmentação das massas cinzenta e branca, e com avanço na correção de não-homogeneidade da intensidade ao longo do cérebro.

# Referências Bibliográficas

- [1] A. X. Falcão, J. K. Udupa, and F. K. Miyazawa. An ultra-fast user-steered image segmentation paradigm: Live-wire-on-the-fly. *IEEE Transactions on Medical Imaging*, 19(1):55–62, Jan 2000.
- [2] R. A. Lotufo, A. X. Falcão, and F. Zampirolli. IFT-Watershed from gray-scale marker. In *Proc. of XV Brazilian Symp. on Computer Graphics and Image Processing*, pages 146–152. IEEE, Oct 2002.
- [3] S. P. Awate, T. Tasdizen, N. Foster, and R. T. Whitaker. Adaptive markov modeling for mutual-information-based, unsupervised MRI brain-tissue classification. *Medical Image Analysis*, 10(5):726–39, 2006.
- [4] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley*, pages 281–297. University of California Press, 1967.
- [5] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.
- [6] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, 1981.
- [7] A. K. Jain, R. P. W. Duin, and J. Mao. Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37, 2000.
- [8] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):790–799, Aug 1995.
- [9] M. Herbin, N. Bonnet, and P. Vautrot. A clustering method based on the estimation of the probability density function and on the skeleton by influence zones. In *Proceedings of the Pattern Recognition Letters*, volume 17, pages 1141–1150, 1996.

- [10] D. Comaniciu and P. Meer. A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:603–619, 2002.
- [11] D. DeMenthon. Spatio-temporal segmentation of video by hierarchical mean shift analysis. In *Proceedings of Statistical Methods in Video Processing Workshop*, 2002.
- [12] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *IEEE Conference on Computer Vision and Pattern*, pages 142–151, 2000.
- [13] D. Comaniciu and P. Meer. Kernel-based object tracking. In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, volume 25, pages 564–577. IEEE Computer Society, May 2003.
- [14] J. Wang, B. Thiesson, Y. Xu, and M. Cohen. Image and video segmentation by anisotropic kernel mean shift. In *Proc. of the 8th European Conference on Computer Vision*, volume 3022, pages 238–249. Springer Berlin / Heidelberg, 2004.
- [15] Changjiang Yang, Ramani Duraiswami, and Larry Davis. Efficient mean-shift tracking via a new similarity measure. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 176–183, Washington, DC, USA, 2005. IEEE Computer Society.
- [16] E. Parzen. On the estimation of a probability density function and mode. *Annals of Mathematical Statistics*, 33:1065–1076, 1962.
- [17] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience, 2 edition, 2000.
- [18] Z. Wu and R. Leahy. An optimal graph theoretic approach to data clustering: theory and its applications to image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(11):1101–1113, Nov 1993.
- [19] A. X. Falcão, J. Stolfi, and R. A. Lotufo. The image foresting transform: Theory, algorithms, and applications. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(1):19–29, 2004.
- [20] A. X. Falcão, B. S. da Cunha, and R. A. Lotufo. Design of connected operators using the image foresting transform. In *Proc. of SPIE on Medical Imaging*, volume 4322, pages 468–479, Feb 2001.
- [21] L. Vincent. Morphological grayscale reconstruction in image analysis. *IEEE Transactions on Image Processing*, 2(2):176–201, Apr 1993.

- [22] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, New York, 1973.
- [23] Keinosuke Fukunaga. *Introduction to statistical pattern recognition (2nd ed. )*. Academic Press Professional, Inc., San Diego, CA, USA, 1990.
- [24] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264–323, September 1999.
- [25] L. J. Hubert. Some applications of graph theory to clustering. *Psychometrika*, 39(3):283–309, 1974.
- [26] G. Ball and D. Hall. Isodata: A novel method of data analysis and pattern classification. Technical report, Stanford Research Institute, Menlo Park, 1965.
- [27] A. K. Jain and R. C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.
- [28] I. J. Cox, S. B. Rao, and Y. Zhong. Ratio regions: a technique for image segmentation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 557–564, 1996.
- [29] Song Wang and Jeffrey Mark Siskind. Image segmentation with minimum mean cut. In *Intl. Conf. on Computer Vision (ICCV)*, volume 1, pages 517–525, Jul 2001.
- [30] S. Sarkar and K. L. Boyer. Quantitative measures of change based on feature organization: eigenvalues and eigenvectors. In *Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 478–483, 1996.
- [31] Jiambo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):888–905, Aug 2000.
- [32] Song Wang and Jeffrey Mark Sinkind. Image segmentation with ratio cut. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(6):675–690, Jun 2003.
- [33] Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. In *International Conference on Computer Vision (ICCV)*, volume 1, pages 105–112, 2001.
- [34] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, Sep 2004.

- [35] Vladimir Kolmogorov and Ramin Zabih. What energy functions can be minimized via graph cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(2):147–159, Feb 2004.
- [36] C. Fowlkes, S. Belongie, and J. Malik. Efficient spatiotemporal grouping using the nyström method. In *Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 231–238, 2001.
- [37] Julio Carballido Gamio, Serge J. Belongie, and Sharmila Majumdar. Normalized cuts in 3D for spinal MRI segmentation. *IEEE Trans. on Medical Imaging*, 23(1):36–44, Jan 2004.
- [38] L. Ford and D. Fulkerson. *Flows in networks*. Princeton University Press, 1962.
- [39] D. Greig, B. Porteous, and A. Seheult. Exact maximum a posteriori estimation for binary images. *J. Royal Statistical Society, series B*, 51(2):271–279, 1989.
- [40] A. X. Falcao, P. A. V. Miranda, and A. Rocha. A linear-time approach for image segmentation using graph-cut measures. In *Advanced Concepts for Intelligent Vision Systems*, pages 138–149, Sep 2006.
- [41] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Trans. on Information Theory*, 21:32–40, Jan 1975.
- [42] Jue Wang, Pravin Bhat, R. Alex Colburn, Maneesh Agrawala, and Michael F. Cohen. Interactive video cutout. *ACM Trans. Graph.*, 24(3):585–594, 2005.
- [43] V. Katkovnik and I. Shmulevich. Nonparametric density estimation with adaptive varying window size. In *Proc. of the Conf. on Image and Signal Processing for Remote Sensing*, pages 25–29, 2000.
- [44] D. Comaniciu, V. Ramesh, and P. Meer. The variable bandwidth mean shift and data-driven scale selection. In *Proc. of the Eighth IEEE International Conference on Computer Vision*, volume 1, pages 438–445, 2001.
- [45] R. Bellman. *Dynamic Programming*. Princeton University, 1957.
- [46] E. F. Moore. The shortest path through a maze. In *Proc. Intl. Symp. on the Theory of Switching*, pages 285–292. Harvard University, Apr 1959.
- [47] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

- [48] R. B. Dial. Shortest-path forest with topological ordering. *Comm. of the ACM*, 12(11):632–633, Nov 1969.
- [49] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network flows: theory, algorithms, and applications*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.
- [50] R. A. Lotufo and A. X. Falcão. The ordered queue and the optimality of the watershed approaches. In *Mathematical Morphology and its Applications to Image and Signal Processing*, volume 18, pages 341–350. Kluwer, Jun 2000.
- [51] J. K. Udupa and S. Samarasekera. Fuzzy connectedness and object definition: Theory, algorithms, and applications in image segmentation. *Graphical Models and Image Processing*, 58:246–261, 1996.
- [52] P. K. Saha and J. K. Udupa. Fuzzy connected object delineation: Axiomatic path strength definition and the case of multiple seeds. *Computer Vision and Image Understanding*, 83:275–295, 2001.
- [53] L. G. Nyul, A. X. Falcão, and J. K. Udupa. Fuzzy-connected 3d image segmentation at interactive speeds. *Graphical Models*, 64:259–281, September 2002.
- [54] B. S. Cunha. Projeto de operadores de processamento e análise de imagens baseados na transformada imagem-floresta. Master’s thesis, Universidade Estadual de Campinas, Instituto de Computação, Ago 2001.
- [55] A. X. Falcão, B. S. da Cunha, and R. A. Lotufo. Design of connected operators using the image foresting transform. In *SPIE on Medical Imaging*, volume 4322, pages 468–479, Feb 2001.
- [56] R. Audigier and R. A. Lotufo. Seed-relative segmentation robustness of watershed and fuzzy connectedness approaches. In *XX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI)*, pages 61–68, Belo Horizonte, MG, Oct 2007. IEEE CPS.
- [57] R. Audigier and R. A. Lotufo. Watershed by image foresting transform, tie-zone, and theoretical relationship with other watershed definitions. In *Mathematical Morphology and its Applications to Signal and Image Processing (ISMM)*, pages 277–288, Rio de Janeiro, RJ, Oct 2007. MCT/INPE.
- [58] F. Cappabianco, G. Araújo, and A. X. Falcão. The image foresting transform architecture. In *IEEE International Conference on Field Programmable Technology (ICFPT)*, Kokurakita, Kitakyushu, Japan, Dec 2007. IEEE.

- [59] A. X. Falcão and F. P. G. Bergo. Interactive volume segmentation with differential image foresting transforms. *IEEE Trans. on Medical Imaging*, 23(9):1100–1108, 2004.
- [60] F. P. G. Bergo and A. X. Falcão. A partitioned algorithm for the image foresting transform. In *Mathematical Morphology and its Applications to Signal and Image Processing (ISMM)*, pages 425–436, Rio de Janeiro, RJ, Oct 2007. MCT/INPE.
- [61] P. Salembier and J. Serra. Flat zones filtering, connected operators, and filters by reconstruction. *IEEE Trans. on Image Processing*, 4(8):1153–1160, Aug 1995.
- [62] S. Beucher and F. Meyer. The morphological approach to segmentation: The watershed transformation. In *Mathematical Morphology in Image Processing*, chapter 12, pages 433–481. Marcel Dekker, 1993.
- [63] J. B. T. M. Roerdink and A. Meijster. The watershed transform: Definitions, algorithms and parallelization strategies. *Fundamenta Informaticae*, 41:187–228, 2000.
- [64] T. Cormen, C. Leiserson, and R. Rivest. *Introduction to Algorithms*. MIT, 1990.
- [65] L. Vincent and P. Soille. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(6), Jun 1991.
- [66] R. Audigier and R. A. Lotufo. Seed-relative segmentation robustness of watershed and fuzzy connectedness approaches. In *Proc. of the 20th Brazilian Symposium on Computer Graphics and Image Processing*, pages 61–68. IEEE Computer Society, 2007.
- [67] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, Nov 2001.
- [68] J. A. Garcia, J. Fdez-Valdivia, F. J. Cortijo, and R. Molina. A dynamic approach for clustering data. *Signal Processing*, 44(2):181–196, 1995.
- [69] J. Cutrona, N. Bonnet, and M. Herbin. A new fuzzy clustering technique based on pdf estimation. In *Proceedings of Information Processing and Managing of Uncertainty*, pages 225–232, 2002.
- [70] D. Comaniciu. An algorithm for data-driven bandwidth selection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(2):281–288, 2003.

- [71] S. Paris and F. Durand. A topological approach to hierarchical segmentation using mean shift. In *CVPR*, pages 18–23. IEEE Computer Society, June 2007.
- [72] J. P. Papa, A. X. Falcão, P. A. V. Miranda, C. T. N. Suzuki, and N. D. A. Mascarenhas. Design of robust pattern classifiers based on optimum-path forests. In *Mathematical Morphology and its Applications to Signal and Image Processing (ISMM)*, pages 337–348. MCT/INPE, 2007.
- [73] J. P. Papa, A. X. Falcão, C. T. N. Suzuki, and N. D. A. Mascarenhas. A discrete approach for supervised pattern recognition. In *12th International Workshop on Combinatorial Image Analysis*, volume 4958, pages 136–147. Springer Berlin/Heidelberg, 2008.
- [74] J. A. Montoya-Zegarra, J. P. Papa, N. J. Leite, R. S. Torres, and A. X. Falcão. Rotation-invariant texture recognition. In *3rd International Symposium on Visual Computing*, volume Part II, LNCS 4842, pages 193–204, Lake Tahoe, Nevada, CA, USA, Nov 2007. Springer.
- [75] J. A. Montoya-Zegarra, J. P. Papa, N. J. Leite, R. S. Torres, and A. X. Falcão. Learning how to extract rotation-invariant and scale-invariant features from texture images. In *Journal on Advances in Signal Processing*, 2008. (aceito para publicação).
- [76] A. A. Spadotto, J. C. Pereira, R. C. Guido, J. P. Papa, A. X. Falcão, A. R. Gatto, P. C. Cola, and A. O. Shelp. Oropharyngeal dysphagia identification using wavelets and optimum path forest. In *Proceedings of the 3th IEEE International Symposium on Communications, Control and Signal Processing*, pages 735–740, 2008. ISBN: 978-1-4244-1688-2.
- [77] J. P. Papa, A. A. Spadotto, A. X. Falcão, and J. C. Pereira. Optimum path forest classifier applied to laryngeal pathology detection. In *15th International Conference on Systems, Signals and Image Processing*, 2008. (aceito para publicação).
- [78] A. X. Falcão, C. T. N. Suzuki, J. F. Gomes, J. P. Papa, L. Candido, and S. Shimizu. Sistema para diagnóstico de parasitos intestinais por análise computadorizada de imagens. Protocolo INPI - PI0605465-0 (pedido solicitado).
- [79] A. X. Falcão, C. T. N. Suzuki, J. F. Gomes, J. P. Papa, L. Candido, and S. Shimizu. A system for diagnosing intestinal parasites by computerized image analysis. Protocolo PCT - PCT/BR2007/000272 (pedido solicitado).
- [80] L. Vincent. Morphological area opening and closings for greyscale images. In *Shape in Picture '92 - NATO Workshop*. Springer, Sep 1992.
- [81] S. Beucher and C. Lantuejoul. Use of watersheds in contour detection. In *Proceedings of the International Workshop on Image Processing, Real-Time Edge and Motion Detection*, 1979.

- [82] F. Meyer. Topographic distance and watershed lines. *Signal Processing*, 38:113–125, 1994.
- [83] P. K. Saha, J. K. Udupa, and D. Odhner. Scale-based fuzzy connected image segmentation: Theory, algorithms, and validation. *Computer Vision and Image Understanding*, 77(2):145–174, 2000.
- [84] B. Georgescu, I. Shimshoni, and P. Meer. Mean shift based clustering in high dimensions: A texture classification example. In *Proc. of the Intl. Conf. on Computer Vision*, pages 456–463. IEEE Computer Society, 2003.
- [85] D. O. Loftsgaarden and C. P. Quesenberry. A nonparametric estimate of a multivariate density function. *The Annals of Mathematical Statistics*, 36(3):1049–1051, June 1965.
- [86] S. Sharma. *Applied multivariate techniques*. John Wiley & Sons Inc., New York, NY, USA, 1996.
- [87] S. Theodoridis and K. Koutroubas. *Pattern Recognition*. Academic Press, New York, NU, USA, 1999.
- [88] M. Halkidi, M. Vazirgiannis, and Y. Batistakis. Quality scheme assessment in the clustering process. In *Proc. of the 4th European Conf. on Principles of Data Mining and Knowledge Discovery*, 2000.
- [89] M. Halkidi and M. Vazirgiannis. Clustering validity assessment: Finding the optimal partitioning of a data set. In *Proc. of the IEEE Intl. Conf. on Data Mining*, pages 187–194, 2001.
- [90] A. X. Falcão and J. K. Udupa. A 3D generalization of user-steered live wire segmentation. *Medical Imaging Analysis*, 4(4):389–402, Dec 2000.
- [91] L. M. Rocha, A. X. Falcão, and L. G. P. Meloni. A robust extension of the mean shift algorithm using optimum path forest. In *Proc. of the 12th Intl. Workshop on Combinatorial Image Analysis*, 2008.
- [92] D. Lowe. Distinctive image features from scale-invariant keypoints. In *Proc. of the International Journal of Computer Vision*, volume 20, pages 91–110, 2003.
- [93] T. Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, 21:224–270, 1994.
- [94] E. Dougherty and R. A. Lotufo. *Hands-on Morphology Image Processing*. International Society for Optical Engineering, 2003.

- [95] O. Cuisenaire and B. Macq. Fast k-NN Classification with an Optimal k-Distance Transformation Algorithm. In *Proc. 10th European Signal Processing Conference*, Med. Physics, pages 1365–1368. SPIE, 2000.
- [96] MPEG-7. Mpeg-7: The generic multimedia content description standard, part 1. *IEEE Multi-Media*, 09(2):78–87, 2002.
- [97] D.J. Newman A. Asuncion. UCI machine learning repository, 2007.
- [98] N. Arica and F. T. Y. Vural. BAS: A Perceptual Shape Descriptor Based on the Beam Angle Statistics. *Pattern Recognition Letters*, 24(9-10):1627–1639, June 2003.
- [99] E. Persoon and K. Fu. Shape Discrimination Using Fourier Descriptors. *IEEE Transactions on Systems, Man, and Cybernetics*, 7(3):170–178, 1977.
- [100] R. Torres, A. X. Falcão, and L.F. Costa. A graph-based approach for multiscale shape analysis. *Pattern Recognition*, 37(6):1163–1174, 2004.
- [101] P. Salembier and L. Garrido. Connected operators based on region-tree pruning strategies. In *15th Intl. Conf. on Pattern Recognition*, volume 03, page 3371. IEEE Computer Society, 2000.
- [102] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.
- [103] L. M. Rocha, F. A. M. Cappabianco, and A. X. Falcão. Data clustering as an optimum-path forest problem with applications in image analysis. *International Journal of Imaging Systems and Technology*, 2008. submetido.
- [104] P. A. V. Miranda, F. P. G. Bergo, L. M. Rocha, and A. X. Falcão. Tree-pruning: A new algorithm and its comparative analysis with the watershed transform for automatic image segmentation. In *XIX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI)*, pages 37–44. IEEE, Oct 2006.
- [105] F. P. G. Bergo, A. X. Falcão, P. A. V. Miranda, and L. M. Rocha. Automatic image segmentation by tree pruning. *Journal of Mathematical Imaging and Vision*, 29(2–3):141–162, Nov 2007.
- [106] J. P. Papa and A. X. Falcão. A new optimum-path forest approach for supervised classification. In *19th International Conference on Pattern Recognition*, 2008. (submetido).

- 
- [107] D. Lacerda. Recuperação de imagens utilizando a floresta de caminhos ótimos. Master's thesis, Universidade Estadual de Campinas, Instituto de Computação, 2008.
- [108] L. F. Pinto. Recuperação de imagens utilizando a floresta de caminhos ótimos. Master's thesis, Universidade Estadual de Campinas, Instituto de Computação, 2008.
- [109] F. A. M. Cappabianco, A. X. Falcão, and L. M. Rocha. Clustering by optimum path forest and its application to automatic gm/wm classification in mr-t1 images of the brain. In *Proceedings of the 5th IEEE International Symposium on Biomedical Imaging*, 2008. (aceito para publicação).