

Michelle Matos Horta

***Sistema para Segmentação de Imagens por
Agrupamento Hierárquico baseado no
Comportamento Superparamagnético do Modelo
de Potts***

Recife

março/2004

Michelle Matos Horta

***Sistema para Segmentação de Imagens por
Agrupamento Hierárquico baseado no
Comportamento Superparamagnético do Modelo
de Potts***

Este trabalho foi apresentado à pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para a obtenção do grau de mestre em Ciência da Computação.

Orientador:

Prof. Dr. Alejandro C. Frery

PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DE PERNAMBUCO

Recife

março/2004

Agradecimentos

Este documento é necessário para agradecer as pessoas que me ajudaram neste trabalho e, que muitas vezes, na correria do dia a dia, esquecemos de agradecer. Então, estes agradecimentos de maneira documentada passam a ser uma forma de singela de retribuição.

Em primeiro lugar, agradeço a Deus pelas oportunidades oferecidas e pela vida que tenho.

Em segundo lugar e de forma especial, eu agradeço aos meus pais amados. Eles me proporcionam uma vida maravilhosa, com amor, respeito, educação e apoio em tudo na minha vida. Eu sou uma pessoa de bem e feliz por causa deles.

Em terceiro lugar, eu agradeço aos meus queridos e “eternos” orientadores. No mestrado, o Alejandro C. Frery. Aprendi muito com ele, tanto no lado profissional quanto pessoal. Ele foi mais que um orientador, ele foi e continuará sendo um amigo. Na graduação, o Robson C. Lins que foi um orientador dedicado e importantíssimo na minha formação científica.

Eu também agradeço aos meus amigos, tanto aqueles que conheci na UFPE quanto na UNICAP. Eles sempre me deram apoio de forma direta ou indireta através de favores ou, simplesmente, dando atenções e carinhos.

Dedicatória

Aos meus pais William e Gleice.

Resumo

Recentemente foi estabelecida uma analogia entre o problema de agrupamento e a procura de configurações típicas de um modelo físico, o modelo de Potts. Para ilustrar essa analogia foi desenvolvido um programa, em linguagem C, pelo Professor Eytan Domany, do Departamento de Física de Sistemas Complexos, do *Weizmann Institute of Science*, de Israel. A função principal desse programa é a realização do agrupamento hierárquico de dados baseado no comportamento superparamagnético do modelo de Potts (SPC – *Superparamagnetic Clustering*). Essa coleção de agrupamentos é obtida simulando ocorrências de um modelo de Potts não-homogêneo em um estágio intermediário entre as fases ferromagnética e paramagnética. O foco deste trabalho é o desenvolvimento de um sistema que permite a aplicação dessa técnica à segmentação de imagens. Para tanto, foi desenvolvida uma interface, utilizando IDL e ENVI, para auxiliar o usuário na utilização dessa e de outras técnicas de agrupamento e na compreensão dos resultados provenientes de técnicas hierárquicas de agrupamento de dados. Os agrupamentos obtidos são ilustrados através de imagens, de gráficos hierárquicos (dendrogramas) interativos e por animações.

Palavras-chave: Segmentação de Imagens, Agrupamento Hierárquico de Dados, Modelo de Potts, Interface.

Abstract

Recently an analogy was established between the problem of clustering data and the search of typical configurations of a physical model, the Potts model. To illustrate this analogy, a program was developed, in the C language, by Professor Eytan Domany (Department of Physics of Complex Systems, Weizmann Institute of Science, Israel). The main purpose of this program is the accomplishment of a hierarchical clustering algorithm based on the superparamagnetic behavior of the Potts model (SPC – Superparamagnetic Clustering), through finding typical states of a non-homogeneous Potts model in intermediate states between the ferromagnetic and the paramagnetic phases. This work is aimed at clustering technique to image segmentation. An interface, using IDL and ENVI, was developed in order to assist the user in the use of this and other segmentation procedures and to provide tools to understand the results from hierarchical clustering techniques. Clusters are illustrated through images, interactive hierarchical graphs (dendrogram) and animations.

Keywords: Image Segmentation, Hierarchical Clustering Data, Potts Model, Interface.

Sumário

Lista de Figuras

Lista de Tabelas

1	Introdução	13
1.1	Segmentação de imagens	15
1.2	Agrupamento superparamagnético	18
1.3	Plataformas de trabalho: IDL e ENVI	20
1.3.1	Plataforma IDL	20
1.3.2	Plataforma ENVI	20
1.4	Objetivos do trabalho	21
1.5	Contribuições e estrutura do documento	22
2	Conceitos Básicos	24
2.1	Imagens	24
2.1.1	Definição	24
2.1.2	Formalização	26
2.1.3	Classificações	27
2.1.4	Vizinhança	29
2.2	Padrão de Armazenamento do ENVI	30
2.3	Medidas de Proximidade	32
2.4	Agrupamento de Dados	33
2.5	Agrupamento por Partição	35
2.5.1	k -médias	35
2.6	Agrupamento Hierárquico	37
2.6.1	Métodos Aglomerativos	38
	Algoritmo de Junção Simples	39
	Algoritmo de Junção por Centróide	40

2.6.2	Métodos Divisivos	41
2.7	Aplicação em Imagens	43
3	Agrupamento Superparamagnético	50
3.1	Definição	50
3.2	O Modelo de Ising	51
3.3	O Modelo de Potts	53
3.4	O Método de Swendsen-Wang	54
3.5	Visão Geral do Algoritmo de Agrupamento de Dados	57
3.5.1	Analogia física com o problema de <i>spins</i> de Potts	60
	Associação dos <i>Spins</i> de Potts a cada ponto ω_i	60
	Identificação dos vizinhos	60
	Cálculo das interações locais	62
3.5.2	Localização da fase superparamagnética	62
3.5.3	Determinação de medidas na fase superparamagnética	63
	A Correlação <i>Spin-Spin</i>	63
	Os grupos de dados	64
3.6	Aplicação do Algoritmo SPC a Imagens	64
4	O Domínio da Aplicação	66
4.1	Segmentação de Imagens	66
4.1.1	Definição	67
4.1.2	Problemas	69
4.2	Aplicação do algoritmo superparamagnético à segmentação de imagens	71
4.3	Trabalhos Relacionados	72
5	SPC - Programa e Interface	74
5.1	O Programa SPC	74
5.1.1	Estrutura dos Arquivos de Entrada	75
5.1.2	Estrutura dos Arquivos de Saída	77
5.2	Otimizações	81
5.2.1	Arquivos de Entrada	81
5.2.2	Arquivos de Saída	85

5.2.3	Segmentação de Imagens	87
5.3	Interface	88
5.3.1	Análise e Projeto dos Requisitos	89
	Abrir Imagem	91
	Fechar Imagem	93
	Visualizar Imagem	93
	Fornecer MDA	94
	Gerar MD	95
	Selecionar Algoritmo Agrupamento	96
	Visualizar Imagem Segmentada	97
	Gerar Dendrograma	98
	Procurar Grupo	100
	Gerar Animações	101
5.3.2	Implementação da Interface	102
	Módulo Principal	102
	Módulo MDA	108
	Módulo Imagem Segmentada	114
	Módulo Dendrograma	115
	Módulo Procura Grupo	117
	Módulo Animação	118
6	Resultados	121
6.1	Descrição da Imagem	121
6.2	Resultados	121
6.2.1	Imagem Reduzida	122
6.2.2	Primeira Região da Imagem	124
6.2.3	Segunda Região da Imagem	125
7	Conclusões	128
	Referências Bibliográficas	130

Lista de Figuras

1	Agrupamento de oito objetos caracterizados por medidas bidimensionais.	14
2	Tipos de vizinhança na representação matricial	29
3	Resultados das técnicas aglomerativas e divisíveis.	37
4	Dendrograma para o exemplo da junção simples.	40
5	Representação gráfica de três medidas de dissimilaridade.	42
6	Imagem monocromática.	44
7	Imagem colorida.	45
8	Grade Bidimensional.	51
9	Associação entre <i>spins</i> e estados	55
10	Escolha do <i>spin</i> inicial	55
11	<i>Spins</i> que satisfazem a condição $\zeta_u = \zeta_v$	56
12	Estabelecimento de arco entre <i>spins</i> - Passo 3	57
13	Configuração com grupos de SW	58
14	Nova configuração gerada	59
15	Associação de estados do modelo de Potts a <i>Spins</i>	61
16	Diferenciação nas características da imagens.	70
17	Domínio espacial \times domínio das características.	71
18	Estrutura original do SPC.EXE.	75
19	Imagem monocromática 5×5	79
20	Imagem resultante definida por números.	81
21	Caso de uso geral para o Sistema Proposto.	91
22	Caso de uso detalhado do sistema.	92
23	Diagrama de Resposta ao Evento “Abrir Imagem”.	93
24	Diagrama de Resposta ao Evento “Visualizar Imagem”.	93
25	Diagrama de Resposta ao Evento “Fornecer MDA”.	94
26	Diagrama de Resposta ao Evento “Gerar MD”.	95
27	Diagrama de Resposta ao Evento “Selecionar Algoritmo Agrupamento”.	96

28	Diagrama de Resposta ao Evento “Visualizar Imagem Segmentada”.	97
29	Diagrama de Resposta ao Evento “Gerar Dendrograma”.	99
30	Diagrama de Resposta ao Evento “Procurar Grupo”.	100
31	Diagrama de Resposta ao Evento “Gerar Animações”.	101
32	Tela Principal da Interface.	103
33	Opção <i>File</i> do Menu Principal.	103
34	Opção <i>Options</i> do Menu Principal.	104
35	Resultados da Opção <i>Close</i> do Módulo Principal.	105
36	Opção <i>Show Image</i> do Módulo Principal.	105
37	Opção <i>Dissimilarity Matrix</i> do Módulo Principal.	106
38	Opção <i>Clustering Algoritms</i> do Módulo Principal.	107
39	Módulo MDA.	111
40	Janela <i>Brushplot</i> do Módulo MDA.	112
41	Janela de análise descritiva do Módulo MDA.	113
42	Módulo Imagem Segmentada.	114
43	Tela do Módulo Dendrograma.	115
44	Janelas do Módulo Dendrograma.	117
45	Opção de interação com o dendrograma.	118
46	Janela do Módulo Procura Grupo.	118
47	Janelas do Módulo Animação.	120
48	Ilustração geral das imagens de análise	122
49	Imagens Segmentadas da Imagem de Teste Reduzida	123
50	Ilustração da região 1.	125
51	Imagens segmentadas da região 1 pelo critério de vizinhança-8	126
52	Imagens segmentadas da região 1 pelo critério KNN	126
53	Ilustração da região 2.	126
54	Imagens segmentadas da região 2	127

Lista de Tabelas

1	Exemplo de um arquivo .HDR.	46
2	Matriz de dissimilaridade D_1	47
3	Matriz de dissimilaridade D_2	47
4	Matriz de dissimilaridade D_3	47
5	Exemplo de dados de entrada para o algoritmo de junção por centróide .	47
6	Matriz de dissimilaridade C_1	47
7	Matriz de dissimilaridade C_2	47
8	Matriz de dissimilaridade C_3	48
9	Exemplo de conjunto de dados com variáveis binárias.	48
10	Conjunto de dados da imagem monocromática.	48
11	Matriz de dissimilaridade da imagem monocromática.	48
12	Conjunto de dados da imagem colorida.	48
13	Matriz de dissimilaridade da imagem colorida.	49
14	Arquivo de dados com os dados reais.	76
15	Arquivo de dados com a matriz de dissimilaridade.	76
16	Campos do arquivo com extensão .RUN.	77
17	Exemplo de arquivo com extensão .PARAM.	78
18	Arquivo com extensão .RUN da imagem monocromática 5×5	79
19	Arquivo com extensão DG_01.	80
20	Arquivo com extensão .LAB.	80
21	Matriz de dissimilaridade triangular.	82
22	Matriz de dissimilaridade com critério de vizinhança-4.	83
23	Matriz de dissimilaridade triangular com critério de vizinhança-4.	83
24	Arquivo de dados otimizado.	84
25	Exemplo de arquivo de dados otimizado.	85
26	Arquivo com extensão .LAB com ordem invertida.	86
27	Arquivo com extensão .LAB adaptado.	86

28	Arquivo com extensão DG_01 alterado.	87
29	Principais eventos do sistema.	90
30	Sub-funções da função “Fornecer MDA”.	95
31	Sub-funções da função “Gerar Dendrogramas”.	98
32	Sub-funções da função “Gerar Animações”.	101
33	Recurso IDL para salvar imagem com extensão .EPS.	110
34	Dados de entrada utilizados nas imagens de teste.	122

1 *Introdução*

É intuitivo ao ser humano olhar para o mundo em volta e, em instantes, distinguir os principais objetos contidos nele. Este fato se deve à capacidade do sistema olho-cérebro humano em organizar características presentes na cena de acordo com algum critério de “semelhança”. Esta organização de características permite o reconhecimento dos objetos, uma vez particionada a cena.

Apesar do sistema olho-cérebro humano ter evoluído ao longo de milênios, ele está muito bem adaptado apenas para efetuar a organização de características numéricas de até três dimensões que estejam espacialmente bem separadas.

As imagens, tal como as percebem os seres humanos, podem ser caracterizadas por funções definidas sobre um domínio com valores em um espaço real de dimensão três. Desta forma, o reconhecimento de objetos estáticos pela visão está, para os seres humanos, relacionado com problemas de tratamento de sinais de dimensão três (estamos excluindo a componente temporal).

O reconhecimento de objetos em imagens pode ser simples desde que haja poucos objetos e desde que eles sejam visualmente distintos e bem definidos. Por outro lado, em imagens com muitas informações, como imagens de sensoriamento remoto multiespectrais ou hiperespectrais, com dimensões superiores a três, organizar as características pode se tornar um processo exaustivo, lento, ou até, impossível.

As pesquisas no processamento de imagens e em visão computacional têm fornecido várias ferramentas para auxílio na análise e manipulação de imagens. O processamento de imagens não consegue competir atualmente com o nosso sistema visual, mas pode tornar a manipulação de imagens mais rápida e fácil, como no caso de imagens de sensoriamento remoto.

Segundo Lucca (1998), um dos primeiros passos do processo de análise de uma imagem consiste em particioná-la em seus elementos constituintes (objetos) e representá-la através de um conjunto de regiões obedecendo algum critério para realizar essa partição. Esse critério poderá ser qualquer medida que o usuário considere adequada à aplicação e aos dados, sendo alguns deles a intensidade média, o contraste e a textura. Este processo é denominado “segmentação”.

A segmentação é uma das tarefas mais críticas no processo de análise de uma

imagem e sua importância tem impulsionado o desenvolvimento de uma grande variedade de algoritmos para esse propósito como, por exemplo, os que utilizam técnicas de agrupamento de dados.

O agrupamento de dados consiste em, a partir de um conjunto de dados que caracterizam populações de indivíduos, procurar grupos de elementos semelhantes entre si e distintos dos outros. Alguns dos textos que tratam este problema detalhadamente são os de Duda & Hart (1973), de Duda, Hart & Stork (2001), de Fukunaga (1990), de Kaufman & Rousseeuw (1990) e de Ripley (1996).

Como ilustração para a operação de agrupamento definida acima consideremos um problema simples: agrupar oito indivíduos que estão caracterizados por uma medida bi-dimensional, ou seja, por uma medida formada pela coleta de dois atributos como, por exemplo, estatura e peso. A Figura 1 ilustra esta caracterização mapeando os oito indivíduos em pontos no plano. É intuitivo afirmar que os indivíduos formam dois grupos distintos de objetos, $G1 = \{A, B, C, D\}$ e $G2 = \{E, F, G, H\}$.

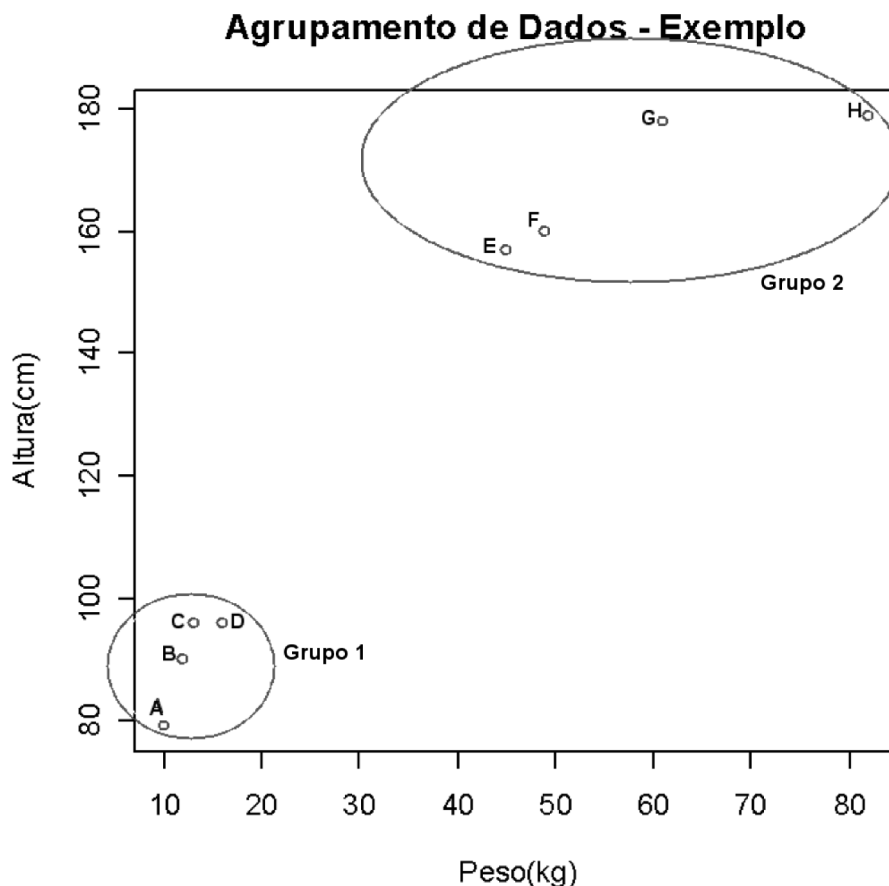


Figura 1: Agrupamento de oito objetos caracterizados por medidas bidimensionais.

Os objetos que pertencem ao mesmo grupo apresentam a característica *altura-peso* com valores próximos entre si, sendo assim temos a formação de dois grupos

(denominados de *clusters* em inglês). A descoberta destes *clusters* é o principal objetivo da *análise de agrupamento*, onde objetos que pertencem ao mesmo grupo são similares entre si e distintos de objetos que pertencem a grupos diferentes.

Pode-se fazer uma analogia entre o problema de agrupar dados à dificuldade do ser humano em aprender a distinguir objetos. Como exemplo, considere uma criança que nunca tenha visto gatos e cachorros. A esta criança foram entregues centenas de fotos desses animais, sem nenhuma explicação prévia dos objetos contidos nas fotos. Há chances da criança ter uma noção de que ela foi apresentada a dois novos tipos de animais, após ter visto uma grande quantidade de informação, ou seja, a criança aprendeu mais dois tipos de animais sem ter sido instruída a isto. Esta forma de aprendizado não-supervisionado é realizada pelo processo contínuo de melhoramento dos esquemas de classificação do subconsciente.

Nosso cérebro aprende, sem uma necessidade de instruções prévias detalhadas, através dos agrupamentos ou junções de observações similares. Tal processo é usado no agrupamento de dados, onde são procurados grupos que têm a característica de serem construídos da forma mais natural possível à percepção humana. Estes grupos são denominados de “grupos naturais”.

Como visto anteriormente, a segmentação de imagens tem o objetivo de particionar os dados de entrada em objetos de acordo com algum critério. Se estes critérios estiverem voltados às características/particularidades das imagens de forma natural à percepção humana, mais a segmentação tende a ter bons resultados do ponto de vista da visão.

As técnicas de agrupamento de dados possuem vantagens com relação a outras técnicas para a questão de análise de características naturais das imagens, visto que podem ser aprimoradas para a obtenção dos “grupos naturais”.

O objetivo desta dissertação é aplicar uma nova técnica de agrupamento no problema de segmentação de imagens, o agrupamento baseado no comportamento superparamagnético do modelo de Potts. Para simplificar a referência a esta técnica, ela será chamada de “agrupamento superparamagnético” em todo este documento. As próximas seções descrevem brevemente os principais assuntos abordados nessa dissertação.

1.1 Segmentação de imagens

A segmentação de imagens é um dos processos mais importantes na análise de imagens. Como a segmentação é um dos primeiros passos na cadeia de operações na análise de uma imagem, o seu desempenho influenciará diretamente na qualidade dos resultados obtidos em etapas posteriores.

O problema central da segmentação é distinguir objetos em uma imagem através

da partição da imagem em regiões disjuntas e homogêneas segundo algum critério estabelecido, tal como textura uniforme, intensidade média e outros. Frequentemente, na imagem segmentada, os objetos encontrados são exibidos através seus contornos.

Como a segmentação é um passo crítico na análise de imagens, várias técnicas foram desenvolvidas através de perspectivas variadas. Em Sharma (2001), tem-se uma revisão bibliográfica dessas técnicas, que podem ser classificadas em pelo menos uma das seguintes categorias largamente utilizadas:

- fatiamento de histograma
- agrupamento de dados
- ajuste de regiões
- por crescimento de regiões
- detecção de bordas

As técnicas baseadas no fatiamento de histograma são frequentemente utilizadas em imagens monocromáticas. Quando estas possuem seus objetos e fundos (*background*) bem definidos, os histogramas apresentam uma estrutura formada por picos e vales. Através do histograma, pode-se estabelecer limiares para separar os alvos distintos da imagem. O histograma é então “fatiado” por estes limiares. Com imagens coloridas, a identificação dos limiares torna-se um processo mais complicado, pois a análise deve ser realizada através de três histogramas, um para cada componente ou banda. A presença de ruídos na imagens dificulta a segmentação por esta técnica, já que este tipo de contaminação irá afetar os histogramas e, com isso, poderá comprometer o resultado obtido. Uma forma de evitar este problema é utilizando filtros para suavizar os ruídos presentes no histograma. Os detalhamento de algumas destas técnicas encontra-se em Sharma (2001) e em Lucchese & Mitra (2001).

As técnicas baseadas nos métodos de agrupamento de dados tem o intuito de particionar a imagem em subgrupos a partir de medidas de proximidade. Assim, os pontos de uma imagem que estão no mesmo grupo são próximos perante esta medida e pontos pertencentes a grupos diferentes são distantes. Estas técnicas, como comentado na introdução deste capítulo, tendem a formar “grupos naturais”. Esta vantagem é importante quando aplicada à segmentação, pois encontra objetos mais adaptados à percepção humana. Porém, os métodos de agrupamento de dados tornam-se computacionalmente pouco eficientes para tratar grandes volumes de dados, como no caso das imagens. Algumas dessas técnicas estão detalhadas no Capítulo 2 e em Sharma (2001).

As técnicas baseadas no ajuste de regiões são realizadas através de tentativas de identificar um padrão para cada porção da imagem de entrada. Uma das mais simples implementações desta abordagem é a técnica de divisão e agrupamento de

regiões (*split and merge*). Neste caso, a segmentação é realizada juntando-se regiões adjacentes se as mesmas forem similares ou dividindo uma região se a mesma não é homogênea. Uma abordagem para segmentar uma imagem através desta técnica consiste em subdividir a imagem de entrada sucessivamente em quadrantes que satisfaçam certa propriedade. Desta forma, para cada quadrante, verifica-se se a propriedade é válida ou não. Se um dado quadrante não satisfizer tal propriedade, ele é subdividido em subquadrantes e assim por diante. O próximo passo é agrupar regiões adjacentes que são similares, isto é, que satisfazem a propriedade definida. O processo termina quando não for mais possível nenhuma divisão ou agrupamento.

Entretanto, existem algumas limitações no emprego desta técnica. Uma delas é que, uma vez que a segmentação é feita a partir da produção de divisões quadradas da imagem original, a segmentação final tende a ter regiões com formas quadradas. Outro aspecto é que as regiões produzidas são dependentes do ponto de partida e regiões homogêneas podem ser segmentadas dependendo da sua posição espacial. Finalmente, esta abordagem tende a perder pequenas regiões dentro de grandes áreas uniformes (Lucca 1998).

As técnicas de crescimento de regiões baseiam-se na procura direta de regiões, onde o procedimento consiste em agrupar pixels ou sub-regiões em regiões maiores. Nesta técnica, a segmentação se inicia com partições pequenas, distintas e homogêneas da imagem para depois agrupar essas partições empregando-se critérios de similaridade de uma certa propriedade (nível médio de cinza, por exemplo) entre partições vizinhas. A seleção do critério de similaridade depende da aplicação considerada e do tipo de dado disponível. Alguns problemas desta técnica são a seleção das partições iniciais, a seleção de propriedades convenientes para crescer (ou agrupar) essas partições e a formulação de uma regra de término para o processo. Outro problema é que toda vez que um novo pixel é unido a um segmento, a característica deste segmento é redefinida. A adição de um pixel a um segmento é dependente do limiar de similaridade automático ou manual adotado. Pixels situados em zonas de fronteira entre duas regiões distintas podem ter características impuras, ou seja, seus valores radiométricos podem ser uma combinação das radiometrias que definem essas regiões. Deste modo, se o limiar estabelecido é muito alto, o crescimento de regiões vai deixar muitos desses pixels como segmentos diminutos. Entretanto, se o limiar é baixo, segmentos que representam classes distintas serão agrupados (Lucca 1998).

As técnicas de detecção de bordas baseiam-se em encontrar limites entre regiões homogêneas de diferentes propriedades, e necessitam de operadores específicos que realcem as bordas antes da segmentação. Uma borda é o limite entre duas regiões com propriedades de níveis de cinza relativamente distintas, e sua detecção pode ser feita automaticamente por operadores de derivadas locais denominados de gradiente, os quais são implementados usando-se janelas (máscaras) de variados tamanhos.

Técnicas de detecção de bordas têm sido largamente empregadas para segmentar

imagens ópticas. Alguns exemplos de operadores de bordas usados em tais imagens são os gradientes de Robert, Prewitt e Sobel. Esses operadores são tipicamente empregados sobre janelas quadradas de lado 3, 5 ou 7, já que o emprego de máscaras maiores pode facilmente introduzir problemas na orientação das bordas (Lucca 1998).

Esta dissertação trabalhou com a aplicação de uma técnica de agrupamento de dados à segmentação de imagens, definida na próxima seção.

1.2 Agrupamento superparamagnético

O problema do agrupamento de dados, descrito no Capítulo 1, levou ao desenvolvimento de técnicas e implementações que sempre tiveram como objetivo principal a corretude dos resultados obtidos. Por uma técnica “correta” entende-se, neste contexto, aquela que produz como saída os “grupos naturais” presentes nos dados de entrada.

Descobrir a qual grupo um determinado indivíduo pertence não é uma tarefa simples, uma vez que existem diversas técnicas para agrupamento de dados com resultados potencialmente diferentes (Everitt, Landau & Leese 2001, Gordon 1999, Kaufman & Rousseeuw 1990).

Algumas dessas técnicas exigem um conhecimento mais profundo do conjunto de dados a ser agrupado, isto por causa da quantidade de parâmetros que algumas técnicas necessitam para que o agrupamento possa ser realizado satisfatoriamente. Nem sempre poderemos fornecer todos esses parâmetros, logo, o processo de agrupamento poderá ficar comprometido e os resultados obtidos poderão não representar a realidade do conjunto de dados.

Descobrir qual é a melhor técnica equivale a perguntar se os agrupamentos resultantes estão realmente corretos, pergunta que nem sempre sabemos responder. As diversas abordagens mais formais ao problema de agrupamento têm, dentre outros objetivos, a procura de métricas capazes de medir a “qualidade” de agrupamentos de dados.

Com base neste problema foi desenvolvido um novo método para agrupamento de dados baseado não apenas em características estatísticas e matemáticas como são os métodos clássicos, mas também em propriedades físicas. Esse método é conhecido como **Agrupamento Superparamagnético** e foi desenvolvido por Domany, Blatt, Gdalyahu & Weinshall (1999).

A técnica de agrupamento superparamagnético é baseada em um modelo físico-magnético, o modelo de *Potts não-homogêneo*, que procura associar configurações deste modelo, baseado na temperatura, a grupos de dados. Essa associação é a chave do método de agrupamento superparamagnético e será descrita no Capítulo 3.

Para verificar a eficiência do método superparamagnético foi implementado um sistema que realiza o agrupamento superparamagnético de dados. Esse programa foi desenvolvido em linguagem C e recebe como parâmetros de entrada um *arquivo de dados* e um *arquivo contendo informações a respeito desses dados* como, por exemplo, o número de indivíduos contidos no arquivo de dados, a temperatura a partir da qual os agrupamentos começam a ser realizados, a temperatura final dos agrupamentos, entre outros que serão descritos na Seção 5.1.

A passagem desses parâmetros de entrada é feita pelo próprio usuário que, inicialmente, tem a inconveniência de gerá-los em algum software matemático a parte, ou de informá-los em um arquivo com formatação padrão. Quando os dados são uma imagem, em particular, esta passagem é bastante inconveniente. Este mecanismo de interação torna a aplicação do sistema ao problema de agrupamento de imagens muito difícil, ou até impossível. Outra dificuldade encontrada foi o uso de linhas de comando como forma de ativar o sistema, o que não é uma tarefa tão simples para a maioria dos usuários que tenham a pretensão de utilizar o programa, uma vez que muitos não pertencem à área de Computação.

Como mencionado no parágrafo anterior, o programa descrito, nomeado SPC, não oferece mecanismos amigáveis de interação ao usuário; a única maneira de interagir com ele é pela passagem de parâmetros através de um arquivo de configuração antes de sua execução. Os arquivos resultantes dos agrupamentos são de difícil entendimento por parte de qualquer usuário, o que também restringe o uso do novo método apenas aos desenvolvedores ou especialistas no assunto.

Procurando amenizar os problemas de interação acima apontados, foi desenvolvida uma interface que realiza a comunicação entre o usuário e o SPC, proporcionando ao usuário não só a passagem dos parâmetros de entrada para o programa, mas também a criação dos mesmos.

Com esta proposta, o usuário pode criar os parâmetros de entrada dentro da própria interface e escolher as informações apropriadas para a realização dos agrupamentos. Além disso, apenas dois dos cinco arquivos de saída do SPC foram considerados para exibição dos resultados, os quais contêm informações que são utilizadas para representar de inúmeras maneiras os agrupamentos.

Enquanto as tarefas acima eram desenvolvidas, foram feitas modificações substanciais ao código original com o objetivo de aperfeiçoar as funcionalidades já existentes e de ajustá-las às necessidades da interface desenvolvida. Para tanto, foi preciso “enxugar” o aglomerado de informações que eram geradas e armazenadas pelo programa original como, também, gerar outros dados relevantes para as novas funções da interface desenvolvida. Além de serem realizadas alterações para adaptá-lo à segmentação de imagens.

Uma descrição mais detalhada da interface desenvolvida é descrita na Seção 5.3. Esta interface foi implementada utilizando IDL e ENVI, plataformas que serão descritas

na próxima seção.

1.3 Plataformas de trabalho: IDL e ENVI

1.3.1 Plataforma IDL

O IDL (*Interactive Data Language*) é um ambiente de trabalho completo para análise e visualização interativa de dados de qualquer natureza (*IDL: Interactive Data Language 2003*). Este software é desenvolvido, mantido e comercializado pela firma RSI (*Research Systems Inc*), que integra o poder de uma linguagem de quarta geração com ferramentas avançadas de análise matemático-estatístico, representação gráfica e visualização em forma de funções e rotinas facilmente acessíveis ao usuário.

Com o IDL se tem grande flexibilidade, podendo os dados serem processados inteiramente no ambiente do IDL através de seus comandos, ou aplicações completas serem desenvolvidas pela linguagem IDL, agregar também rotinas Fortran ou C, ou ainda, chamar funções do IDL num programa Fortran ou C, com livre escolha da plataforma de trabalho.

Outras vantagens do IDL incluem:

- Os operadores e funções/procedimentos podem trabalhar diretamente com matrizes, sem a utilização de laços, o que permite simplificações na iteração e redução no tempo de execução do programa.
- Permite desenhar de forma multidimensional, visualizar imagens, montar animações.
- Contêm rotinas numéricas e estatísticas para análise e simulação dos dados.
- Oferece flexibilidade na leitura e escrita de arquivos em vários formatos: BMP, GEO TIFF, Interfile, JPEG, PICT, PNG, PPM, SRF, TIFF, X11 Bitmap, XWD, CDF, HDF, NetCDF, ASCII, Binary, DICOM, DXF, WAV e XDR.
- Contêm componentes gráficas para criação de interfaces multiplataformas.
- Suporta a execução dos programas para várias plataformas com pouca ou nenhuma modificação nos programas: UNIX, Macintosh e Microsoft Windows.

1.3.2 Plataforma ENVI

O ENVI (*Environment for Visualizing Images*) é um software comercial para processamento de dados de Sensoriamento Remoto, mas que provê a visualização e análise de imagens de virtualmente qualquer tamanho e em inúmeros formatos (*ENVI: Environment for the Visualization of Images 2003*). Seu fabricante, a RSI (*Research Systems*

Inc), o desenvolveu a partir da linguagem IDL. Isso lhe garante robustez, velocidade e sofisticação sem necessitar de equipamento poderoso.

O ENVI está sendo comercializado de duas formas: com, e sem, a linguagem de desenvolvimento IDL. O ENVI RT (versão sem IDL) dispõe de todas as funcionalidades somente através de sua interface. A versão com o IDL torna o ENVI uma arquitetura aberta e permite que seus procedimentos complexos sejam utilizados em modo “batch”, ou seja, sem uma interação com o usuário. Além disso, esta versão permite integrar ao ENVI aplicações que foram desenvolvidas no IDL.

A versão do ENVI com o IDL foi escolhida como plataforma de trabalho, visto que as virtudes mencionadas atendem plenamente as necessidades do trabalho.

1.4 Objetivos do trabalho

A proposta inicial era aplicar o programa SPC à segmentação de imagens e analisar seus resultados. Visto as necessidades definidas na Seção 1.2, foi desenvolvida uma interface gráfica amigável para se trabalhar com imagens. Nela, dois arquivos de entrada são construídos a partir da imagem analisada, um contendo sempre uma matriz de dissimilaridade e o outro informações a respeito dessa matriz. Esses dois arquivos são os parâmetros de entrada do programa SPC, como mencionado na Seção 1.2. Além da geração desses arquivos, um dos objetivos da interface é a comunicação com o próprio SPC, ou seja, visualmente o usuário poderá invocar o programa sem a necessidade de usar linhas de comando.

Após a realização dos agrupamentos, outro objetivo importante é permitir ao usuário uma fácil compreensão dos resultados, através de gráficos, imagens e animações. A interface implementa os gráficos conhecidos como dendrogramas, que se apresentam sob a forma de árvore hierárquica, e permitem a interação do usuário com os agrupamentos realizados. Além disso, o usuário pode visualizar as imagens segmentadas de várias maneiras: por busca, através de animações e outros. Outras funcionalidades da interface são a geração de uma análise multivariada de dados e uma análise descritiva. Dessa forma, os problemas mencionados na Seção 1.2 seriam reduzidos, uma vez que o ambiente gráfico permite uma fácil compreensão das funções fornecidas pelo programa e baseadas nas técnicas utilizadas.

A interface foi desenvolvida com IDL e ENVI e recebe como entrada as imagens do tipo .IMG, que são o padrão de armazenamento do ENVI. Este padrão será descrito no decorrer do texto.

1.5 Contribuições e estrutura do documento

Muitas técnicas de agrupamento de dados tem sido desenvolvidas para diversas aplicações (ver, por exemplo, Duda & Hart 1973, Duda et al. 2001, Fukunaga 1990, Kaufman & Rousseeuw 1990, Ripley 1996), mas todas estas técnicas têm como objetivo a procura de “ grupos naturais” em função das características dos dados de entrada.

Como a técnica de agrupamento superparamagnético visa tratar volumes de dados grandes em uma diversidade de situações, ela pode ser de grande utilidade na segmentação de imagens.

Apesar da técnica de agrupamento superparamagnético ter sido proposta para tratar dados com enormes quantidades de informação, as simulações realizadas para gerar configurações do modelo de Potts são custosas computacionalmente quando aplicadas a dados como imagens, por exemplo. Em Ferber & Wörgötter (2000), tem-se uma otimização deste processo visando a segmentação de imagens de forma rápida e robusta. A otimização foi realizada baseando-se em sistemas biológicos.

Nesta dissertação, o agrupamento superparamagnético original foi adaptado para ser aplicado à segmentação de imagens. As adaptações basearam-se nas peculiaridades da segmentação de imagens, em particular no uso da noção de vizinhança para definir métricas no espaço de atributos. Estas modificações foram realizadas tanto na construção das matrizes de dissimilaridades necessárias ao algoritmo, quanto em um dos passos do mesmo.

Uma interface amigável foi implementada para facilitar as análises em todo o processo, como descrito na seção anterior. Uma das vantagens da ferramenta implementada é a sua flexibilidade no acréscimo de qualquer algoritmo hierárquico. Além disso, a interface está integrada ao software ENVI, que é de uso muito disseminado na comunidade de processamento de imagens.

Além deste capítulo introdutório, o documento apresenta outros seis: conceitos básicos, agrupamento superparamagnético, domínio de aplicação, SPC-programa e interface, resultados e conclusões.

O Capítulo 2 aborda os conceitos básicos necessários: imagens e sua utilização, o formato de armazenamento de uma imagem utilizado na dissertação, as medidas de proximidade que foram utilizadas neste trabalho, uma visão formal das técnicas de agrupamento de dados, e a aplicação de algoritmo de agrupamento de dados à segmentação de imagens.

O Capítulo 3 explica os conceitos introdutórios e, de forma detalhada, o algoritmo da técnica de agrupamento hierárquico baseado no comportamento superparamagnético do modelo de Potts. Além disso, este capítulo discute sua aplicação à segmentação de imagens.

O Capítulo 4 aborda uma explicação geral sobre os conceitos presentes na segmentação de imagens, envolvendo as definições formais e problemas associados a este tipo de aplicação. Descreve a aplicação do algoritmo superparamagnético à segmentação de imagens e apresenta os trabalhos relacionados a este domínio.

O Capítulo 5 aborda o programa *SFC* em detalhes e apresenta as otimizações realizadas, bem como o projeto e a implementação da interface desenvolvida através de análise de requisitos e implementações.

O Capítulo 6 mostra os resultados aplicados a uma imagem de sensoriamento remoto e o Capítulo 7 apresenta as conclusões obtidas e discute trabalhos futuros.

2 *Conceitos Básicos*

Este capítulo serve de auxílio para o entendimento das noções necessárias à dissertação proposta. Ele expõe alguns conceitos básicos sobre imagens e agrupamento de dados. Além disso, tem a importante tarefa de formalizar tanto os tipos de imagens utilizadas, quanto o conceito de agrupamento. Estas formalizações serão utilizadas em toda a dissertação.

2.1 *Imagens*

Dar uma definição à palavra “imagem” é uma tarefa difícil, principalmente devido à subjetividade associada a ela. Um antigo provérbio chinês já dizia: “Uma imagem contém mais de mil palavras...”. Perante a ciência da computação, pode-se associar a imagem a um arquivo contendo uma imensa quantidade de informações, cujas interpretações, diante de um observador humano, são realizadas de forma global e qualitativa através do sistema visual. Quando aplicada ao universo físico, a percepção de imagem se faz pelo resultado de estímulos luminosos produzidos por um suporte bidimensional, seja esta imagem o resultado de um processo intermediado, como, por exemplo, uma fotografia, ou através da projeção do mundo tridimensional na retina do olho humano.

As próximas subseções fornecem uma noção geral sobre imagens, e alguns de seus conceitos, além de formalizar as imagens consideradas nesta dissertação.

2.1.1 *Definição*

Embora existam vários modelos matemáticos para descrever uma imagem, daremos ênfase ao modelo espacial, que é o mais indicado para aplicações em análise de imagens. A seguir está descrita uma definição de imagens baseando-se em Gomes & Velho (1994).

Uma imagem contínua (não-discreta) é uma aplicação $f: S \rightarrow U$, onde $S \subset \mathbb{R}^3$ é uma superfície e U é um espaço vetorial. Na maioria das aplicações S é um subconjunto no plano, e U é um espaço multidimensional constituído de números reais limitados e não-negativos. No entanto, é conveniente para o desenvolvimento da teoria, que U

seja um espaço vetorial qualquer que, em geral, contém este espaço multidimensional limitado como subespaço. A função f na definição é chamada “função imagem”.

O domínio da função imagem, S , é chamado “suporte”. Como está contido no espaço tridimensional, pode exibir diversas estruturas topológicas como, por exemplo, a de máquinas morfológicas (estrutura hexagonal), superfícies e outros. Mas S , em geral, é um produto cartesiano de intervalos pertencentes a \mathbb{Z} , ou seja, assume a estrutura de uma grade definida por $S = [0, \dots, c - 1] \times [0, \dots, \ell - 1]$, onde c denota o número de colunas e ℓ o número de linhas. Esta estrutura de grade, torna a imagem $f(i, j)$ espacialmente discreta, pois as coordenadas (i, j) de cada ponto variam no conjunto S .

O contradomínio da função imagem, U , pode ser chamado de espaço de atributos ou conjunto dos valores da imagem ou, também, gamute de cores da imagem, por possuir o conjunto dos valores possíveis de f . Para codificar a imagem no computador, deve-se trabalhar com modelos de imagens onde a função imagem f toma valores em um subconjunto discreto do espaço de atributos U . Esse processo de discretizar o U de uma imagem é chamado de “quantização” (ver Gomes & Velho 1994).

Mais geralmente, o conjunto U pode ser definido das seguintes maneiras: $U \subset \mathbb{R}^b$ ou $U \subset \mathbb{C}^b$, onde $b \geq 1$ corresponde ao número de bandas ou número de componentes ou, ainda, o número de atributos da imagem f . Dependendo dos valores assumidos por b , as imagens podem ser classificadas em:

- $b = 1$ indica imagens monoespectrais ou monocromáticas, por possuírem uma banda espectral ou uma “cor”;
- $2 \leq b \leq 10$ indica imagens multiespectrais. Quando $b = 3$, tem-se as imagens coloridas;
- $b \geq 11$ indica imagens hiperespectrais.

O valor de b depende do sistema de imageamento utilizado. Segundo Mascarenhas & Velasco (1989), sistemas de imageamento são sistemas físicos destinados a produzir (ou captar) imagens, como câmeras de TV, sensores a bordo de satélites e outros. Embora o aspecto externo destes sistemas varie muito, eles apresentam várias características em comum. Nestes sistemas, a imagem de uma cena é formada num plano através de lentes. Neste plano são colocados sensores que medem a intensidade da luz incidente. No caso da fotografia, por exemplo, o filme fotográfico é colocado no plano dos sensores.

Dependendo da quantidade de luz incidente, os sensores “responderão” proporcionalmente, seja através de uma reação química, como no caso de um filme fotográfico, seja através de um sinal elétrico, como no caso das células fotoelétricas. A resposta do sensor depende da quantidade de luz incidente como também da frequência da luz. É comum, portanto, descrever a característica do sensor através de uma “curva

de resposta espectral”, que descreve a intensidade da resposta para cada frequência (ou comprimento de onda).

Uma imagem considerada colorida é, na realidade, um conjunto de três imagens obtidas por três sensores com curvas de resposta espectral diferentes (por exemplo, com mais sensibilidade às cores verde, azul e vermelho, respectivamente). A sensação de cor que se experimenta é devido à existência, na retina, destes três tipos de sensores. Quando se tem um único sensor é definida uma imagem monocromática. Em Gomes & Velho (1994), tem-se uma definição mais aprofundada dos fundamentos e sistemas de cor.

De um modo geral, chama-se imagem multiespectral ou imagem hiperespectral à coleção de imagens de uma mesma cena, num mesmo instante, obtida por vários sensores com respostas espectrais diferentes. Quando há apenas uma banda espectral, a imagem é chamada de mono-espectral. Estes tipos de imagens são obtidas a partir de satélites e são utilizadas em sensoriamento remoto.

Vale ressaltar ainda, as definições de imagem digital, pixel e função de visualização. Uma imagem digital é uma imagem do tipo discreta-quantizada, ou seja, que é espacialmente discreta no conjunto S e quantizada no conjunto U . Uma formalização deste tipo de imagem está definida na próxima subseção.

Os pixels (acrônimo de *picture element*) são os elementos que constituem uma imagem digital. Eles são compostos pelas coordenadas da imagem e seus respectivos atributos (valores pertencentes ao espaço de atributos da imagem). Pode-se, então, definir o pixel como sendo um par $((i, j), f(i, j))$, onde f é a função imagem e $(i, j) \in S$.

A função de visualização, considerada por $v: U \rightarrow \{\text{cores}\}$, é uma função que mapeia o espaço de atributos em um espaço de cores. Alguns exemplos são: $v: \{0, 1\} \rightarrow \{P, B\}$, exibindo imagens em preto e branco; $v: [0, 255]^3 \rightarrow \{R, G, B\}$, exibindo imagens coloridas. Os espaços de cores são limitados pelos sistemas de exibição da imagem. Pode-se, por exemplo, ter uma imagem colorida, mas com dispositivo gráfico com espaço de cores limitado a uma dimensão, ou seja, com a função de visualização $v: [0, 255]^3 \rightarrow [0, 255]$. Neste caso, será exibida uma imagem monocromática.

Diante dessas definições básicas, podemos formalizar as imagens digitais a serem trabalhadas, cuja descrição encontra-se na próxima subseção.

2.1.2 Formalização

Como comentado na subseção anterior e segundo Mascarenhas & Velasco (1989), uma imagem digital pode ser obtida a partir de uma imagem contínua. Este processo se faz através de dois passos: um, denominado “amostragem”, consiste em discretizar o domínio de definição da imagem; o outro, “quantização”, discretiza o contradomínio de definição da imagem. Assim, considerando a função imagem f da subseção ante-

rior, temos que tanto o suporte S quanto o contradomínio U devem sofrer, respectivamente, amostragem e quantização em subconjuntos $S' \subset S$ e $U' \subset U$ de tal forma que estes sejam discretos.

Diante desse conceito, definiremos a imagem digital através de duas representações matemáticas: uma como função e outra através de um formato matricial. Estas notações estão formalizadas a seguir e foram baseadas nas definições fornecidas pela Subseção 2.1.1.

Uma imagem digital é uma função f definida sobre um conjunto S' , chamado grade, em um conjunto U' de espaço de atributos, cuja representação é $f: S' \rightarrow U'$. O conjunto S' será considerado como sendo da forma $S' = [0, \dots, c-1] \times [0, \dots, \ell-1]$, isto é, uma grade regular de pontos com c colunas e ℓ linhas. O conjunto U' é um espaço b -dimensional com valores naturais contidos num intervalo $[0, t]$, onde $t \geq 1$ e $b \geq 1$. Assim, para cada ponto $(i, j) \in S'$, tem-se b valores definidos em $f(i, j)$.

Uma imagem digital também pode ser definida no formato matricial, considerando cada dimensão do espaço de atributos b -dimensional U' , como uma matriz de ordem $c \times \ell$. Assim, U' é composto de b matrizes bidimensionais. Para formalizar este conceito, consideremos uma matriz A de ordem $c \times \ell$, sendo $A = (a_{i,j})$ uma das b -dimensões de U' , onde $(i, j) \in S'$. Cada elemento $a_{i,j}$ pertence ao intervalo de valores naturais $[0, t]$, com $t \geq 1$. A imagem digital, então, é uma composição de b matrizes A .

A seguir serão exemplificadas algumas imagens segundo as definições presentes nesta subseção. Algumas delas são utilizadas neste trabalho.

2.1.3 Classificações

De acordo com as definições mostradas na subseções anteriores, pode-se definir os seguintes tipos de imagens:

- binárias
- monocromáticas
- monoespectrais
- multiespectrais ou hiperespectrais
- coloridas com uso de *Look Up Table*

Imagens binárias possuem o gamute com dois níveis, ou seja, são funções do tipo $f: [0, \dots, c-1] \times [0, \dots, \ell-1] \rightarrow \{0, 1\}$. Estas imagens normalmente são visualizadas na forma $v: \{0, 1\} \rightarrow \{\text{Preto}, \text{Branco}\}$, onde, normalmente, a cor preta está associada ao valor 0 e a cor branca ao valor 1.

Imagens monocromáticas possuem o gamute com mais de dois níveis, ou seja, são funções do tipo $f: [0, \dots, c-1] \times [0, \dots, \ell-1] \rightarrow [0, t]$ com $t \geq 2$. Na visualização

das imagens, podem-se distinguir diferentes níveis de intensidade de cinza, obtidos através da quantificação das intensidades de luz intermediárias entre o preto e o branco. O número de tons de cinza t é normalmente uma potência de dois, e depende do número de bits usados para armazenar o valor da intensidade do pixel. Frequentemente trabalha-se com 256 tonalidades, o que equivale a 8 bits/pixel e fornece um intervalo $[0, 255]$. Este número é mais que suficiente para representar as diferentes tonalidades da escala de cinza perceptíveis ao olho humano, mas pode ser insuficiente para armazenar toda a informação disponível na cena e, tal como mostrado por Mascarenhas, Dutra & Frery (2000), valores reduzidos de t podem acarretar na impossibilidade de descrevê-la corretamente. Assim como uma imagem monocromática pode ser visualizada em uma escala de tons de cinza, é possível fazê-la em outras escalas (de vermelho, de verde, de azul e outros.).

A faixa espectral imageada não precisa ficar restrita apenas às frequências visíveis, podendo atingir outras faixas de comprimento de onda como, por exemplo, o infravermelho. Esta banda (representação da faixa de comprimento de onda em Sensoriamento Remoto) é usada em satélites que buscam detectar, por exemplo, a emissão de calor e, conseqüentemente, identificar queimadas. Imagens monoespectrais são constituídas de uma única faixa espectral imageada, ou seja, uma banda ($b = 1$).

Imagens multiespectrais ou hiperespectrais são formadas pela mesma função $f: [0, \dots, c - 1] \times [0, \dots, l - 1] \rightarrow [0, t]^b$. Os valores assumidos por b é que definem qual das duas imagens considerar; em geral, quando $1 < b < 11$ elas são consideradas multiespectrais, caso contrário, são consideradas hiperespectrais. A quantificação das intensidades de mais de uma banda espectral por vez fornecerá uma informação mais completa a respeito da cena adquirida. Este é o caso da obtenção de imagens coloridas, pois estas são obtidas através de sensores específicos para a determinação de cada uma das intensidades de vermelho (*Red*), verde (*Green*) e azul (*Blue*), formando a representação RGB. A mistura destas três bandas ou cores, permite que se tenha a sensação de enxergar imagens coloridas, uma vez que o olho humano possui sensores que atuam justamente nestas três faixas de comprimento de onda. As imagens coloridas são usualmente armazenadas por conjuntos de 24 bits, com 8 bits para representar as intensidades de vermelho, 8 bits para o verde e 8 bits para o azul. Com a composição destas três cores básicas utilizando-se 24 bits/pixel, pode-se chegar a um número de até 16 milhões de cores e tonalidades distintas. Este número é perfeitamente adequado para a representação da realidade, em cenas digitalizadas, sem a menor perda de detalhes e qualidade em relação às cores. Isto é devido ao fato deste número de cores ser em muito superior à capacidade do olho humano em distinguir cores e tonalidades.

A obtenção de imagens coloridas não está restrita apenas às imagens com representação multiespectral para cada ponto. Pode-se gerar uma imagem onde cada pixel é associado a um valor referente a uma tabela de palette (LUT - *Look Up Table*), que descreve a cor real deste ponto da imagem. Desta forma não é preciso associar a cada

ponto da imagem as informações a respeito das intensidades RGB, podendo então armazenar imagens de cenas coloridas sem a necessidade de reservar uma memória de 24 bits/pixel. Para este tipo de imagens, uma cena é composta pelo mapa de pixels, ou seja, a matriz $c \times \ell$ de pontos com os índices de acesso às cores da LUT, adicionada a uma tabela de acesso indireto às cores reais (LUT), a chamada tabela de palette.

2.1.4 Vizinhança

A estrutura de uma imagem no formato matricial determina implicitamente uma relação de conectividade entre os elementos da imagem (Bustos, Frery & Ojeda 1998, Frery, Sant'Anna, Mascarenhas & Bustos 1997, Gomes & Velho 1994). Esta relação necessita do conceito de vizinhança, definido a seguir, considerando as formalizações apresentadas na Subseção 2.1.2.

Dado um elemento $a_{i,j}$ da imagem definida pela matriz A de ordem $c \times \ell$, consideramos os vizinhos de $a_{i,j}$ (vizinhança de $a_{i,j}$) como os elementos mais próximos de $a_{i,j}$. Em particular, a vizinhança-4 de $a_{i,j}$, ou $N_4(a_{i,j})$ é o conjunto de quatro elementos $a_{i-1,j}$, $a_{i+1,j}$, $a_{i,j-1}$ e $a_{i,j+1}$, ilustrada na Figura 2(a) e a vizinhança-8 de $a_{i,j}$, ou $N_8(a_{i,j})$, o conjunto de elementos da vizinhança-4 mais $a_{i-1,j-1}$, $a_{i-1,j+1}$, $a_{i+1,j-1}$ e $a_{i+1,j+1}$, ilustrada na Figura 2(b).

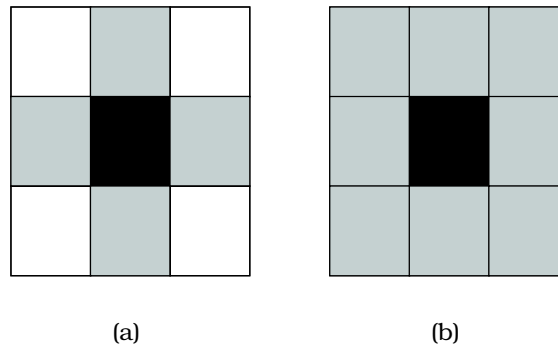


Figura 2: Tipos de vizinhança na representação matricial: (a) vizinhança-4 e (b) vizinhança-8.

Será de interesse medir a distância entre dois pontos $p_1 = (i_1, j_1)$ e $p_2 = (i_2, j_2)$, o que pode ser feito pela distância Euclidiana (ver Seção 2.3):

$$d(p_1, p_2) = \sqrt{(i_1 - i_2)^2 + (j_1 - j_2)^2}.$$

Fazendo uma analogia entre pontos de um plano e os elementos da matriz $A = (a_{i,j})$, estabelecemos que a distância entre o elemento $a_{i,j}$ e seus vizinhos de vizinhança-4, é de 1 unidade. Já a distância entre um elemento $a_{i,j}$ e seus vizinhos de vizinhança-8, que não sejam vizinhança-4, tem $\sqrt{2}$ unidades.

2.2 Padrão de Armazenamento do ENVI

O ENVI utiliza um padrão aberto para armazenar imagens que é constituído por dois arquivos: um contendo os atributos de uma imagem e outro contendo o cabeçalho. Esta estrutura de armazenamento o torna flexível para uma grande variedade de imagens, e garante a portabilidade dos dados gerados. Como o ENVI é um software comercial voltado ao processamento de dados de Sensoriamento Remoto, os conceitos que ele implementa e, conseqüentemente, a forma em que realiza manipulação de imagens, estão atrelados a esta área, ou seja, envolvendo imagens multiespectrais, bandas espectrais e outros. Por outro lado, e em grande medida graças ao padrão aberto de armazenamento de dados empregado, pode-se tratar qualquer tipo de imagem adequando as suas propriedades às do padrão estabelecido pelo ENVI.

Os dados de uma imagem são utilizados pelo ENVI em formato de raster generalizado, ou seja, como se fossem matrizes bidimensionais formadas por pixels. Este formato é armazenado em um arquivo binário, cuja extensão é `.IMG`. Quando há mais de uma banda presente existem três maneiras de organizá-las:

- por bandas inteiras (BSQ, *band sequential*),
- por pixels intercalados (BIP, *band interleaved by pixel*), e
- por linhas intercaladas (BIL, *band interleaved by band*).

O padrão BSQ (bandas em seqüência) é considerado a forma mais simples de formatar imagens em raster. Cada linha é seguida imediatamente pela próxima linha na mesma banda espectral. Assim, percorre-se toda a banda para depois fazer o mesmo na próxima banda. Este formato apresenta os melhores resultados no acesso espacial a qualquer lugar da imagem na mesma banda espectral.

No formato BIP (bandas intercaladas por pixel), armazena-se o primeiro pixel banda por banda até que todos os pixels da primeira posição sejam armazenados. Em seguida, todos os segundos pixels e assim até os últimos pixels da imagem. Este formato possibilita um desempenho ótimo para tratamento espectral aos dados da imagem.

No BIL (bandas intercaladas por linha), armazena-se a primeira linha banda por banda até que todas as primeiras linhas de todas as bandas sejam armazenadas. Em seguida, armazenam-se as segundas linhas e assim até as últimas linhas da imagem. Este formato permite um desempenho intermediário entre os dois formatos apresentados acima.

O arquivo `.IMG` é acompanhado por um pequeno arquivo de cabeçalho em formato de caracteres no padrão ASCII. O arquivo de cabeçalho possui a extensão `.HDR`

(por *header*, “cabeçalho” em inglês) e contem informações necessárias à leitura do arquivo `.IMG`, como as dimensões da imagem, o formato de armazenamento dos dados e outras especificações pertinentes. Esta abordagem torna o arquivo de cabeçalho independente e de fácil manuseio, podendo este ser alterado ou criado a partir de editores de texto. A primeira linha do arquivo `.HDR` deve ter a palavra `ENVI` para que o software o reconheça como seu cabeçalho padrão. As outras informações representam os parâmetros necessários à leitura do arquivo `.IMG` ou para descrever a imagem a ser analisada, algumas delas são:

description: um conjunto de caracteres (texto) que faz uma descrição da imagem analisada ou do processamento aplicado na mesma

samples: número de colunas de cada banda

lines: número de linhas de cada banda

bands: número de bandas na imagem

header offset: número de bytes de cabeçalho embutidos (*embedded*) no arquivo de dados; esses bytes são desprezados quando o arquivo é lido

file type: especifica o tipo de arquivo `ENVI`, se são formato de dados, resultados de processamentos, uma classificação e outros.

data type: identifica o tipo de dado (byte, integer, float, double, e outros)

interleave: identifica o tipo de armazenamento (BSQ, BIP ou BIL)

sensor type: especifica o sensor utilizado (Landsat TM, SPOT, RadarSat, e outros)

byte order: identifica a posição do bit mais significativo na palavra

map info: lista as coordenadas geográficas na projeção UTM

z plot range: taxa de valores mínimos e máximos da plotagem em Z

z plot titles: permite entradas para os eixos X e Y para a plotagem em Z

band names: permite entradas para nomear as bandas da imagem

wavelength: lista os centros dos comprimentos de onda de cada banda da imagem.

Um exemplo comum do arquivo de cabeçalho é ilustrado na Tabela 1 (página 46).

Neste trabalho será empregada exclusivamente a plataforma `ENVI`, dada sua extensibilidade e as facilidades oferecidas (ver Seção 1.3.2).

As próximas quatro seções fornecem um resumo dos conteúdos necessários para entender o conceito de agrupamento de dados. Estas seções são baseadas em Oliveira (2004). Em seguida, é explicado a aplicação de agrupamento de dados à segmentação de imagens.

2.3 Medidas de Proximidade

O esforço para produzir uma simples estrutura de agrupamento a partir de um conjunto de dados complexos requer necessariamente uma medida de “proximidade”, que é usualmente indicada por alguma ordem de distância. Esta medida define o quão próximos estão os dados do conjunto e pode ser classificada em duas categorias: similaridade ou dissimilaridade. A escolha de uma medida de proximidade envolve um pouco de subjetividade, mas algumas considerações importantes incluem: a natureza das variáveis (discreta, contínua, binária), escalas de medida (nominal, ordinal, intervalar, proporção) e o conhecimento do assunto.

A maioria dos métodos de agrupamento assumem que as relações em um conjunto de n objetos são descritas através de uma matriz de tamanho $n \times n$ contendo uma medida de similaridade s_{ij} ou dissimilaridade d_{ij} entre o i -ésimo e j -ésimo objeto para cada par de objetos (i, j) , onde $i = j = 1, \dots, n$. Assim, pode-se dizer, que dois indivíduos estão “próximos” quando sua dissimilaridade d_{ij} é pequena, ou sua similaridade s_{ij} é grande.

Existe uma grande diversidade de medidas de proximidade, mas nossa atenção será restrita às medidas de dissimilaridade. Como estas medidas definem objetos próximos quando seus valores são mínimos, elas são associadas às métricas de distância, cujas propriedades, entre pares de objetos de d_{ij} , satisfazem as seguintes condições:

1. a função é não negativa: $d_{ij} \geq 0$;
2. a função é nula apenas quando avaliada no mesmo elemento: $d_{ii} = 0$;
3. a função é simétrica: $d_{ij} = d_{ji}$.

Uma matriz de dissimilaridade $\mathcal{D} = (d_{ij})_{1 \leq i, j \leq n}$ é considerada uma métrica se satisfaz a desigualdade triangular, $d_{ij} \leq d_{ik} + d_{kj}$, para todos os grupos de três indivíduos $1 \leq i, j, k \leq n$.

Uma variedade de medidas já foram propostas para obter uma matriz de dissimilaridade de um conjunto de observações multivariadas contínuas. As medidas mais comuns são: a distância Euclidiana, a distância de Manhattan, distância de Minkowski (que generaliza as duas anteriores) e a distância de Canberra. Em Gower (1985) e em Gower & Legendre (1986), encontram-se maiores informações sobre outras medidas multivariadas contínuas.

A distância Euclidiana é a medida de distância mais comumente utilizada, e pode ser descrita da seguinte forma,

$$d_{ij} = \sqrt{\sum_{k=1}^p (x_{ik} - x_{jk})^2},$$

onde x_{ik} e x_{jk} são, respectivamente, o k -ésimo valor da variável para os indivíduos i e j , no espaço p -dimensional. Esta medida apresenta uma propriedade interessante: d_{ij} pode ser interpretado como a distância física entre dois pontos $X_i = (x_{i1}, x_{i2}, \dots, x_{ip})$ e $X_j = (x_{j1}, x_{j2}, \dots, x_{jp})$ no espaço Euclidiano de dimensão p .

A distância Manhattan,

$$d_{ij} = \sum_{k=1}^p |x_{ik} - x_{jk}|,$$

descreve uma configuração retilínea, ou seja, todas as distâncias serão medidas em linha reta (Larson & Sadiq 1983).

Uma outra medida de distância é a métrica de Minkowski,

$$d_{ij} = \left(\sum_{k=1}^p |x_{ik} - x_{jk}|^r \right)^{1/r},$$

com $r \geq 1$. Pode-se perceber que as distâncias Euclidiana e Manhattan são casos especiais desta métrica para $r = 2$ e $r = 1$, respectivamente.

Finalmente, a distância de Canberra, que é dada por

$$d_{ij} = \begin{cases} 0 & x_{ik} = x_{jk} = 0 \\ \sum_{k=1}^p \frac{|x_{ik} - x_{jk}|}{(|x_{ik}| + |x_{jk}|)} & x_{ik} \neq 0 \text{ ou } x_{jk} \neq 0 \end{cases}$$

é sensível a pequenas mudanças próximas de $x_{ik} = x_{jk} = 0$ (Lance & Willians 1966). Ela é freqüentemente vista como uma generalização de uma medida de dissimilaridade para dados binários (ver Everitt et al. 2001). Dessa forma, a mesma pode ser padronizada, dividindo o seu resultado pelo número de variáveis, p , para assegurar que o coeficiente de dissimilaridade permaneça no intervalo $[0, 1]$.

Vale ressaltar que existem medidas para outros tipo de variáveis, tais como: variáveis binárias, variáveis nominais e ordinais e variáveis mistas, podendo estas serem encontradas em Everitt et al. (2001), Kaufman & Rousseeuw (1990), Gower & Legendre (1986) e Johnson & Wichern (1992).

2.4 Agrupamento de Dados

Como visto no Capítulo 1, o agrupamento de dados é uma das operações mais importantes na análise de informações. Esta operação consiste em, dada uma massa de informações a respeito de uma população de indivíduos, procurar grupos de elementos semelhantes entre si e diferentes dos outros. Uma ilustração dessa operação de agrupamento pode ser vista na Figura 1 (página 14). Alguns dos textos que tratam este problema detalhadamente são Devroye, Györfi & Lugosi (1996), Duda et al. (2001), Fukunaga (1990), Kaufman & Rousseeuw (1990) e Ripley (1996).

A seguir procuraremos formalizar o problema geral de agrupamento. Esta forma-

lização será aplicada ao longo deste documento.

Seja $\Omega = \{\omega_1, \dots, \omega_n\}$ um conjunto finito de indivíduos. A cada indivíduo, $\omega \in \Omega$, estará associado um vetor de atributos \mathbf{x}_ω de dimensão $p \geq 1$, $p \in \mathbb{N}$. Os vetores de atributos $\mathbf{x}_\omega = (x_{\omega,1}, \dots, x_{\omega,p})$, $\omega \in \Omega$, serão as entradas primárias do problema de agrupamento. Como comentado na Seção 2.3, há uma vasta literatura a respeito da utilização simultânea dos valores reais dos p atributos para fazer o agrupamento dos n indivíduos (ver Kaufman & Rousseeuw 1990), mas neste trabalho o foco será colocado em técnicas que empregam medidas de dissimilaridade multivariadas.

Uma vez definida a medida de dissimilaridade de interesse, o problema de agrupamento dos dados consiste em procurar partições de Ω de $g \in \mathbb{N}$ elementos não vazios chamados “grupos”, da forma $\mathcal{P} = (\Omega_1, \dots, \Omega_g)$, onde $\Omega_i \cap \Omega_j = \emptyset$ se $i \neq j$ e $\cup \Omega_i = \Omega$, para as quais são satisfeitas simultaneamente as condições:

1. A variação das dissimilaridades dentro dos grupos é mínima, e
2. A variação das dissimilaridades entre grupos é máxima.

A primeira condição tenta preservar a propriedade de coesão dos indivíduos de um grupo, ou seja, quanto menor a variação das dissimilaridades, mais similares são os indivíduos pertencentes a um mesmo grupo.

A segunda condição está associada à propriedade de isolamento entre grupos, isto é, quanto maior for a variação das dissimilaridades entre grupos, mais distantes eles estarão.

É possível formalizar de diversas maneiras os critérios que os grupos devem satisfazer. Dependendo do critério de agrupamento utilizado na formação de novos grupos, mais ou menos aglomerados podem ser formados. Caso esse critério seja “muito frouxo”, as diferenças entre os indivíduos não são percebidas e nesse caso apenas um grupo contendo todos os indivíduos é formado. Já se o critério utilizado for “muito rigoroso”, as semelhanças entre os indivíduos não são percebidas e assim cada um deles estaria em um único grupo. Bons algoritmos de agrupamento devem oferecer ao usuário o controle do rigor com que o grupos serão compostos.

Para solucionar o problema geral de agrupamento surgem diversos métodos e seus algoritmos. Dependendo das características do conjunto de dados a ser agrupado, vários desses algoritmos podem ser eficientes para um mesmo conjunto de dados, ou todos eles podem apresentar resultados insatisfatórios.

Para realizar uma boa escolha é preciso conhecer o algoritmo que se está utilizando, além de analisar e comparar, com outros algoritmos, os resultados obtidos. Com base nessa necessidade, este capítulo define os principais métodos de agrupamentos de dados e descreve alguns de seus algoritmos, são eles:

- Métodos de agrupamento por partição, que são os mais desenvolvidos e estuda-

dos, por apresentar única resposta a cada saída, tornando-os métodos diretos e práticos de se trabalharem;

- Métodos de agrupamento hierárquico, que apresentam várias soluções, constituindo uma árvore hierárquica. Esta variedade de respostas pode se tornar uma ferramenta poderosa na análise de informações.

2.5 Agrupamento por Partição

Este método de agrupamento tem como objetivo a construção de uma partição $\mathcal{P} = \{\Omega_1, \dots, \Omega_g\}$ com cada Ω_i representando um grupo, onde todos eles devem apresentar as seguintes características:

- Cada grupo deverá conter no mínimo um objeto ($\nexists \Omega_i = \emptyset$)
- Cada objeto deverá pertencer exatamente a um grupo ($\Omega_i \cap \Omega_j = \emptyset$ se $i \neq j$).

Os métodos de agrupamento por partição tipicamente começam o processamento com base em uma partição inicial do conjunto de dados a ser agrupado, e então usam uma estratégia de controle iterativo para otimizar uma função objetivo. Cada grupo pode ser representado, por exemplo, pelo centro de gravidade do grupo (algoritmos baseado em k -médias) ou por um dos objetos do grupo localizado próximo do seu centro (algoritmos baseados em k -medóides).

A otimização da função objetivo é reportada como um “critério de agrupamento”, pois fornece uma idéia intuitiva dos grupos através de uma fórmula matemática (ver Peña, Lozano & Larrañaga 1999). O valor da função usualmente depende da partição atual do conjunto de dados $\mathcal{P}_i = \{\Omega_1, \dots, \Omega_g\}$, isto é:

$$F: P_g(\Omega) \rightarrow \mathbb{R},$$

onde $P_g(\Omega)$ é o conjunto de todas as partições da base de dados $\Omega = \{\omega_1, \dots, \omega_n\}$ em g grupos não vazios. Cada ω_i dos n objetos da base de dados Ω é um vetor p -dimensional.

A seção a seguir explica um dos algoritmos de partição mais utilizados.

2.5.1 k -médias

O Algoritmo de k -médias (k -Means) foi apresentado originalmente por McQueen (1967), e utiliza os centróides de cada grupo como seus pontos representantes. Em uma grande escala de dados estes algoritmos são mais eficientes do que os algoritmos hierárquicos tradicionais, porém seu desempenho dependente da seleção inicial dos centróides (Bradley & Fayyad 1998).

Como a maioria das técnicas de agrupamento, os centróides finais dos grupos não representam uma solução ótima global, mas apenas uma solução ótima local. Além disso, os resultados obtidos podem ser completamente diferentes por causa da variedade de escolhas a serem atribuídas aos centróides iniciais. Para tanto, existem diversas técnicas propostas para a escolha dos centróides, sendo algumas delas aleatórias (ver McQueen 1967).

O algoritmo k -médias começa selecionando uma partição inicial dos dados. Para esta partição inicial, calcula-se um centróide a cada grupo. Em seguida, cada objeto do conjunto de dados é realocado para o grupo que tiver o centro mais próximo em uma tentativa de reduzir a dissimilaridade dentro dos grupos. Se um objeto muda de grupo, por exemplo do Ω_s para o Ω_t , os centróides destes grupos são modificados, sendo necessário atualizar o valor desses centróides e recalculá-los as distâncias entre os objetos a estes novos centróides. Este processo continua até que haja convergência, ou seja, os objetos permanecem fixos em seus grupos. Os principais passos do algoritmo estão descritos a seguir:

1. Selecionar uma partição inicial do conjunto de dados em g grupos $\mathcal{P} = \{\Omega_1, \dots, \Omega_g\}$
2. Calcular os centróides dos grupos, considerando n_k o número total de objetos num grupo Ω_k , tem-se $\bar{\omega}_k = \frac{1}{n_k} \sum_{j=1}^{n_k} \omega_{kj}$, $k = 1, \dots, g$
3. Para todo ω_k e seguindo a seqüência dos casos faça
 - (a) Realoque o objeto ω_k para o grupo com o centróide mais próximo, $\omega_k \in \Omega_s$ é movido de Ω_s para Ω_t se $\|\omega_k - \bar{\omega}_t\| \leq \|\omega_k - \bar{\omega}_s\|$, $\forall j = 1, \dots, g, t \neq s$
 - (b) Recalcule os centróides para os grupos Ω_s e Ω_t
4. Se os membros dos grupos estão estabilizados então pare senão volte ao passo 3,

onde $\bar{\omega}_k$ é a média aritmética calculada em cada grupo, e $\|\omega_k - \bar{\omega}_t\|$ uma distância a ser considerada entre ω_k e $\bar{\omega}_t$, um exemplo é a distância Euclidiana.

Apesar do algoritmo k -médias ser utilizado em uma variedade de aplicações, ele não está isento de desvantagens, algumas das quais tem sido amplamente discutidas na literatura. As mais importantes são listadas abaixo,

- Como muitos métodos de agrupamento, o k -médias requer a especificação do número de grupos g a priori, e, obviamente, esta especificação prévia é impraticável a uma grande variedade de aplicações reais;
- Como uma técnica iterativa, o algoritmo k -médias é especialmente sensível às condições iniciais, grupos iniciais e ordem dos objetos;
- O algoritmo k -médias converge de forma limitada a um mínimo local da função de custo associada.

2.6 Agrupamento Hierárquico

O resultado obtido por um algoritmo hierárquico é um conjunto de partições para os dados de entrada e todos os possíveis valores de $g = \{1, \dots, n\}$, tipicamente organizado na forma de uma árvore. O conjunto de partições será definido como $P = \{\mathcal{P}_1, \dots, \mathcal{P}_n\}$. Quando $g = 1$, tem-se $\mathcal{P}_1 = \{\Omega\}$ sendo $\Omega = \{\omega_1, \dots, \omega_n\}$, ou seja, tem-se uma partição formada por um único grupo contendo todos os n indivíduos. Quando $g = n$, tem-se $\mathcal{P}_n = \{\Omega_1, \dots, \Omega_n\}$ sendo $\Omega_i = \{\omega_i\}$, ou seja, tem-se uma partição formada por n grupos unitários. Os resultados obtidos por valores intermediários, $g = \{2, 3, \dots, n-1\}$, formam uma transição gradual entre esses dois extremos, onde a diferença entre $g = k$ e $g = k+1$ é a divisão de um cluster em dois para se obter $k+1$ grupos. Analisando a operação numa direção oposta, encontra-se a diferença entre $g = k+1$ e $g = k$ através da junção entre dois clusters para se obter k grupos.

Em muitas definições clássicas, a passagem de $g = k$ para $g = k+1$ ($g = k+1$ para $g = k$, respectivamente) se efetua pela divisão (pela fusão, respectivamente) de exatamente um grupo (dois grupos, respectivamente) em dois (em um, respectivamente). O algoritmo SPC, que será explorado neste trabalho, não procede necessariamente dessa forma. Para este algoritmo, o agrupamento hierárquico não requer a especificação da passagem de $g = k$ para $g = k+1$ (nem de $g = k+1$ para $g = k$). No entanto, será mantida esta especificação no restante desta seção introdutória.

A forma com que os resultados são obtidos define dois tipos de técnicas: as aglomerativas, que geram partições \mathcal{P}_i com $g = \{n, \dots, 1\}$ pelo agrupamento de pares de clusters a cada iteração; e as divisivas, que formam partições \mathcal{P}_i com $g = \{1, \dots, n\}$ dividindo a cada etapa um cluster em dois. A Figura 3 (exemplo de Kaufman & Rousseeuw (1990)) ilustra os resultados dos agrupamentos para um conjunto de dados com $n = 5$ indivíduos. As direções seguidas pelos métodos aglomerativo e divisivo encontram-se, respectivamente, na parte superior e inferior. No exemplo, os resultados dos dois tipos de métodos coincidem, mas usualmente eles são diferentes (Kaufman & Rousseeuw 1990). Diferentes também são os seus respectivos custos computacionais, tipicamente.

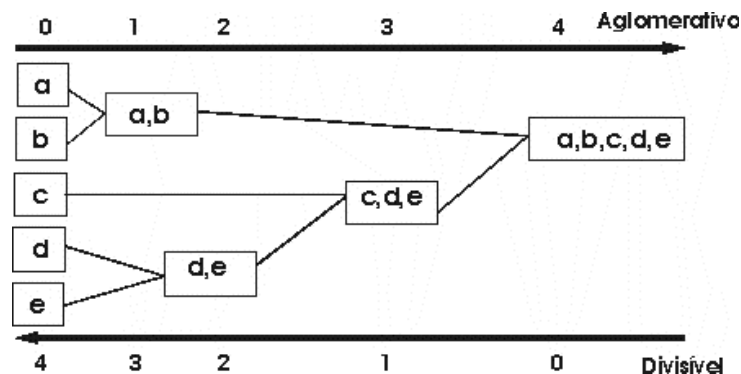


Figura 3: Resultados das técnicas aglomerativas e divisíveis.

As divisões e uniões realizadas pelos métodos hierárquicos são irreversíveis, ou seja, se um algoritmo aglomerativo juntar dois indivíduos, estes não poderão ser separados nas próximas iterações. O mesmo ocorre com os métodos divisíveis, quando ocorre a divisão de grupo em dois, estes não poderão ser unidos novamente. Este fato faz com que uma possível falha nos resultados obtidos em uma iteração anterior nunca seja reparada.

As aplicações típicas dos métodos hierárquicos são na biologia, em especial na construção de árvores taxonômicas, em estudos de sistemas sociais, na biblioteconomia e na arqueologia.

Como ilustrado na Figura 3 (página 37) o conjunto de partições, $P = \{\mathcal{P}_1, \dots, \mathcal{P}_n\}$, pode ser representado por um diagrama bidimensional, chamado “dendrograma”. O dendrograma tem a estrutura de uma árvore: os nós representam os grupos formados; os níveis representam as partições \mathcal{P}_i , $1 \leq i \leq n$, e as folhas estão associadas aos grupos formados pela primeira partição \mathcal{P}_1 . Como exemplo, considere o dendrograma formado pelo método aglomerativo da Figura 3, tem-se as n folhas representando os n grupos unitários de \mathcal{P}_1 , e quatro nós cujas associações são para os grupos: $\{a, b\}$ no nível 1; $\{d, e\}$ no nível 2; $\{c, d, e\}$ no nível 3 e $\{a, b, c, d, e\}$ no nível 4.

Normalmente, os dendrogramas possuem a topologia de árvores binárias por conta das limitações encontradas nas técnicas clássicas. Uma ilustração dessa estrutura pode ser vista na Figura 4 (página 40) mas, como explicado no início dessa seção, sua estrutura pode variar em qualquer tipo de árvore quando aplicada a métodos “generalizados” de agrupamento hierárquico.

Quando o volume de dados é considerável, a construção dos dendrogramas pode se tornar computacionalmente custosa. Além disso, os dendrogramas resultantes podem ser objetos de difícil, ou até impossível, análise visual. Este fato se deve à ordem com que os dados são dispostos nas folhas do dendrograma, que pode influenciar diretamente na análise dos resultados, dado que existem 2^{n-1} formas de construção. Este problema é chamado de “ordenação do dendrograma” (ver Morris, Asnake & Yen 2003).

As duas subseções subsequentes, explicam esses dois tipos de métodos hierárquicos em detalhes, mostrando alguns dos algoritmos que os implementam.

2.6.1 Métodos Aglomerativos

Os métodos aglomerativos são, provavelmente, os mais utilizados entre os métodos hierárquicos. Eles iniciam seus algoritmos com uma partição contendo n grupos unitários, $\mathcal{P}_1 = \{\Omega_1, \dots, \Omega_n\}$, com $\Omega_i = \{\omega_i\}$. A cada iteração dois grupos são unidos, até chegar a uma partição contendo um único grupo que conterá todos os indivíduos, definida por $\mathcal{P}_n = \{\Omega\}$ com $\Omega = \{\omega_1, \dots, \omega_n\}$. O critério de agrupamento utilizado depende das medidas de proximidade calculadas sobre os dados de entrada.

As operações básicas de todos os métodos aglomerativos são similares, diferenciando-se pelas medidas de dissimilaridade (ou similaridade) escolhidas. Assim sendo, só precisam ser explicados em detalhes dois exemplos específicos para ilustrar estas técnicas e, para tanto, foram escolhidos a junção simples (*single linkage*) e a junção por centróide (*centroid linkage*).

Algoritmo de Junção Simples

O algoritmo de *junção simples* também é conhecido como técnica do vizinho mais próximo (*nearest-neighbour*). É um dos métodos hierárquicos mais simples. A definição do método se faz pelas distâncias mais próximas (medidas de dissimilaridades) que formam o critério de união entre dois grupos ou indivíduos. Esta técnica também pode ser aplicada a métodos divisivos, onde o critério para dividir dois grupos utiliza medidas de similaridades (Everitt et al. 2001).

O critério de escolha dos grupos a serem unidos é definido por: dada uma matriz de dissimilaridade (distância) de tamanho $n \times n$ contendo valores representados por d_{ij} , com $I, J \in N$, onde N é um conjunto constituído por todos os indivíduos e subconjuntos obtidos até a iteração anterior, juntar dois grupos ou indivíduos denotados por I e J se e somente se $\min_{\{I, J \in N | I < J\}}(d_{IJ})$.

Uma ilustração deste método encontra-se no exemplo apresentado na Tabela 2 (página 47). Há uma matriz triangular, denominada D_1 , que é formada pelo cálculo de uma medida de dissimilaridade aplicada ao conjunto de indivíduos, definido por $\{1, 2, 3, 4, 5\}$.

No exemplo, os indivíduos escolhidos na fase inicial são 1 e 2, porque possuem o menor valor de dissimilaridade diferente de 0 (zero), no caso $d_{12} = 2$. Redefine-se os valores de dissimilaridades de acordo com os novos dados:

$$\begin{aligned} d_{(12)3} &= \min\{d_{13}, d_{23}\} = d_{23} = 5 \\ d_{(12)4} &= \min\{d_{14}, d_{24}\} = d_{24} = 9 \\ d_{(12)5} &= \min\{d_{15}, d_{25}\} = d_{25} = 8 \end{aligned}$$

Uma nova matriz é definida de acordo com as informações atualizadas. Esta matriz é denominada D_2 e está ilustrada na Tabela 3 (página 47).

De acordo com a matriz D_2 , os indivíduos 4 e 5 são agrupados. Novamente a matriz de dissimilaridade é redefinida de acordo com o grupo formado $\{4, 5\}$. A matriz resultante é a ilustrada na Tabela 4 (página 47).

Em seguida, são agrupados o indivíduo 3 com o grupo $\{4, 5\}$. Finalmente os grupos $\{1, 2\}$ e $\{3, 4, 5\}$ são agrupados em um único grupo. O dendrograma ilustrado na Figura 4 (página 40) retrata o exemplo citado acima. Ele exhibe as partições em cada passo, ou seja, em cada nível da árvore. É importante notar que o dendrograma,

tal como exibido na Figura 4, exibe muita informação: os indivíduos e como eles se agrupam a diversas distâncias.

De uma análise de um dendrograma podem ser extraídas importantes informações a respeito da estrutura dos dados. O dendrograma exibido na Figura 4 permite deduzir, por exemplo, que

- os indivíduos rotulados como 1 e 2 são os mais próximos,
- os indivíduos 4 e 5 são, na ordem, os seguintes mais próximos,
- o indivíduo 3 é parecido com o grupo formado pelos indivíduos 4 e 5,
- que o grupo formado pelos indivíduos 1 e 2 é consistentemente diferente dos grupos formados pelos outros três indivíduos.

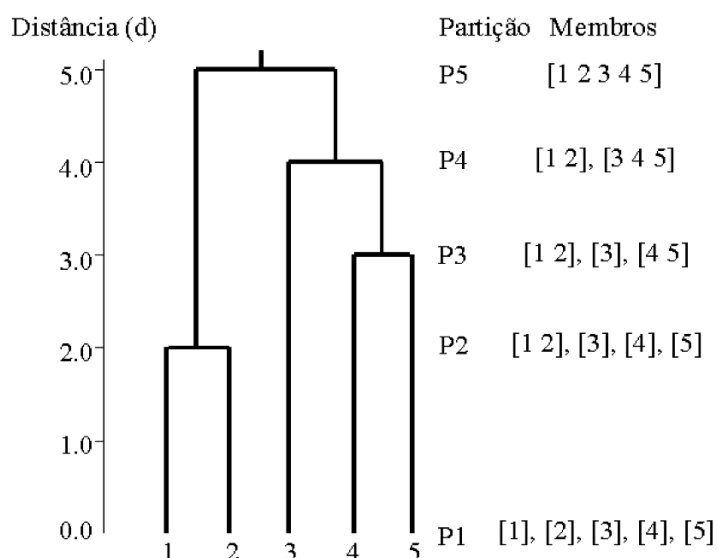


Figura 4: Dendrograma para o exemplo da junção simples.

Algoritmo de Junção por Centróide

O método de junção por centróide, também conhecido como UPGMC (*Unweighted Pair-Group Method using Centroid approach*), baseia-se no método de junção simples, porém ao invés de operar diretamente em uma matriz de dissimilaridade, ele requer o acesso dos dados de entrada. Com a informação aos dados de entrada, calcula-se a matriz de dissimilaridade através de métricas previamente estabelecidas, como a distância Euclidiana.

Em seguida, procura-se pelos grupos/indivíduos que possuam a menor dissimilaridade. Recalcula-se a matriz com as informações obtidas pelo dado de entrada (e não pela matriz de dissimilaridade anterior, como ocorre na junção simples). Este processo é repetido até a definição do grupo final, que conterà todos os n indivíduos.

Uma ilustração do método de junção por centróide é apresentada na Tabela 5 (página 47). Dada uma matriz de $n = 5$ indivíduos contendo duas variáveis. Estabeleceu-se a distância Euclidiana para encontrar as matrizes de dissimilaridade. A partir dos dados de entrada, a matriz de dissimilaridade inicial, denominada C_1 , é mostrada na Tabela 6 (página 47).

Através da matriz C_1 , une-se os indivíduos com menor dissimilaridade, 1 e 2. O vetor médio (centróide) do grupo é calculado, no caso (1, 1.5):

$$\text{centroide}_{12} = ((1.0 + 1.0)/2, (1.0 + 2.0)/2) = (1.0, 1.5)$$

A nova matriz de dissimilaridade será calculada com este novo valor que corresponde ao centróide para o grupo $\{1, 2\}$, ela está ilustrada na Tabela 7 (página 47).

Segundo a matriz C_2 , o próximo grupo a ser formado conterá $\{4, 5\}$. O vetor médio encontrado para este grupo foi (8.0, 1.0). Este vetor influenciará diretamente na próxima matriz ilustrada na Tabela 8 (página 48).

Na matriz C_3 , o indivíduo 3 se agrupa com o grupo $\{4, 5\}$. O estágio final une os dois grupos $\{1, 2\}$ e $\{3, 4, 5\}$.

Existem outros algoritmos aglomerativos que se assemelham a estes dois métodos. O diferencial se encontra nas métricas que definem os critérios de agrupamento. Por exemplo, o método de junção completa (conhecido também como método do vizinho mais distante) tem a idéia oposta ao de junção simples, a medida de proximidade é definida pelas maiores distâncias entre grupos. A junção média, também conhecida como UPGMA (*Unweighted Pair-Group Method using Average approach*), as distâncias entre dois grupos são definidas pelas médias das distâncias entre todos os indivíduos dos dois grupos. Uma ilustração destes dois métodos encontra-se na Figura 5 (figura de Kaufman & Rousseeuw (1990), página 42).

2.6.2 Métodos Divisivos

Diferentemente dos métodos aglomerativos, os métodos divisivos iniciam com uma partição formada por um único grupo contendo todos os indivíduos, $\mathcal{P}_1 = \{\Omega\}$ com $\Omega = \{\omega_1, \dots, \omega_n\}$. A cada iteração, um grupo é dividido em dois outros grupos até ser obtida uma partição formada por n grupos unitários, definida por $\mathcal{P}_n = \{\Omega_1, \dots, \Omega_n\}$, com $\Omega_i = \{\omega_i\}$.

Esses métodos exigem, computacionalmente, no máximo $2^{k-1} - 1$ divisões possíveis em cada nível, ou seja, maneiras de um grupo com k objetos ser dividido em dois subgrupos. Entretanto, para dados binários (dados que possuem apenas dois valores, 0 e 1) existem algoritmos relativamente simples e computacionalmente eficientes. Um

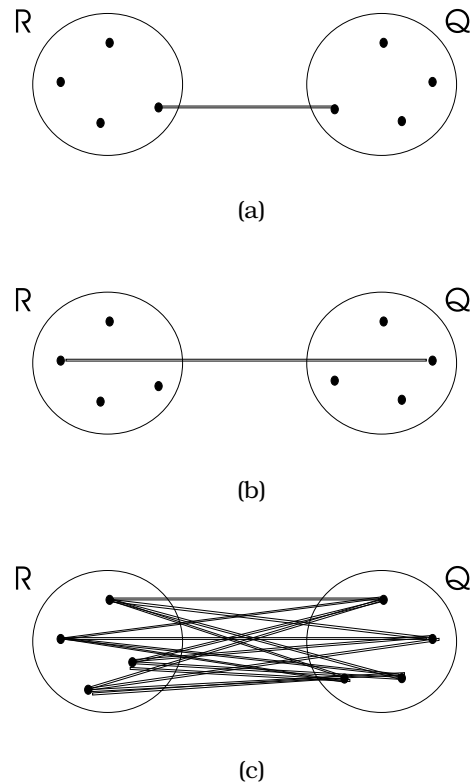


Figura 5: Representação das medidas de dissimilaridade definidas entre os métodos: (a) junção simples; (b) junção Completa; (c) junção média: $d_{RQ} = \sum_{i \in R, j \in Q} d_{i,j}/n$.

exemplo desses algoritmos é o MONA (*Monothetic Analysis*). Este algoritmo é completamente diferente dos vistos até agora, pois sua idéia básica é selecionar uma das variáveis do conjunto de objetos original e dividir este em objetos com e sem o atributo correspondente. Dessa maneira, para cada variável do conjunto de objetos, serão obtidos dois subconjuntos, sendo que o processo continua com a escolha de uma outra variável que, da mesma maneira, divide esse subconjunto em dois grupos menores. Esse processo continua até que o subconjunto contenha um único objeto ou então que as variáveis restantes não consigam mais separar os objetos. Esta última situação só ocorre quando cada variável contém valores constantes para todos os objetos no subconjunto. Para ilustrar essa última situação considere a Tabela 9 (página 48), onde os objetos **B** e **C** não podem ser separados.

Pelo fato de que o conjunto de objetos é dividido em subconjuntos e este processo é contínuo dentro de cada subconjunto, o algoritmo é hierárquico. Mais precisamente ele é divisivo. Além disso, por ser a separação realizada usando uma única variável por vez, ela é chamada de *monotética*.

A parte mais importante do algoritmo é a escolha da variável que irá dividir o conjunto de objetos. A idéia principal é escolher a variável mais centralmente localizada, ou seja, escolher a variável que apresenta a maior soma das “similaridades” a todas as outras variáveis.

Uma maneira simples e utilizada por MONA para obter uma medida de similaridade entre duas variáveis é calcular o produto entre o número de objetos para os quais as duas variáveis possuem o valor 0 e o número para os quais possuem o valor 1. Em seguida, calcular o produto entre o número de objetos para os quais a primeira variável apresenta valor 0 e a segunda 1 e o número de objetos para os quais a primeira variável apresenta valor 1 e a segunda 0. A medida de similaridade é definida como sendo o valor absoluto da diferença entre esses dois produtos. Para uma melhor compreensão do cálculo dessa medida de similaridade considere novamente a Tabela 9 (página 48). As variáveis 1 e 2 são idênticas e dessa forma deveriam apresentar uma alta similaridade. Os dois produtos são 3 (1×3) e 0 (0×0), então a medida de similaridade é $|3 - 0| = 3$. Já as variáveis 1 e 3 são muito diferentes, logo devem apresentar uma baixa similaridade. Os produtos para elas são 0 (0×1) e 2 (1×2). A medida de similaridade para as variáveis 1 e 3 é $|0 - 2| = 2$.

A medida de similaridade apresentada só é recomendada se as duas variáveis utilizadas na análise fornecem divisões similares do conjunto de dados. As variáveis 1 e 2 por ser idênticas não apresentam problemas, mas as variáveis 1 e 3, por exemplo, apresentam informações muito diferentes e mesmo assim o valor da similaridade foi próximo do valor calculado para as variáveis 1 e 2. Logo, pode-se concluir que se duas variáveis apresentam valores diferentes para todos os objetos de um conjunto de dados, elas dão informações idênticas e a similaridade torna-se alta. Como um exemplo desta conclusão considere as variáveis 1 e 4, que são completamente diferentes. Calculando os produtos temos 0 (0×0) e 3 (1×3). A medida de similaridade para as variáveis 1 e 4 é $|0 - 3| = 3$, que é exatamente o mesmo valor obtido quando duas variáveis são iguais.

Como o objetivo é encontrar a variável que é mais similar a todas as outras, a soma das similaridades a todas as outras variáveis é então maximizada. A variável para a qual a soma é máxima, ou seja, a variável que apresentar a maior soma entre todas as calculadas será a escolhida para dividir o conjunto ou subconjunto de dados.

2.7 Aplicação em Imagens

Definidos os conceitos de imagem (Seção 2.1), os de medidas de proximidade (Seção 2.3) e de agrupamento de dados (Seção 2.4), podemos descrever a aplicação dos métodos de agrupamento ao problema de segmentar imagens utilizando medidas de proximidade. Esta descrição será consolidada com exemplos simples: um para imagens monocromáticas e outro para imagens coloridas, ambas em representação matricial.

O problema de agrupar dados consiste em juntar os indivíduos presentes em um conjunto de dados, de acordo com algum critério de proximidade. Quando aplicado a imagens, o conjunto de dados é a própria imagem, sendo este conjunto formado

pelos elementos que compõem a imagem, ou seja, os pixels da imagem, definidos pelo par “posição-valor na posição” $((i, j), f(i, j))$. Assim sendo, os pixels passam a ser os indivíduos (ω_i) do conjunto $\Omega = \{\omega_1, \dots, \omega_n\}$ de dados, sendo o número de observações o tamanho da imagem, isto é, $n = c\ell$, com c colunas e ℓ linhas. O vetor de atributos $\mathbf{x}_\omega = (x_{\omega_1}, \dots, x_{\omega_p})$ é composto pelos valores assumidos em $f(i, j)$. A dimensão do vetor de atributos corresponde ao número de atributos (bandas) b da imagem. Neste caso o p do agrupamento é definido pelo b da imagem.

Uma distinção importante a fazer quando se realiza segmentação de imagens é que, diferentemente do problema de agrupamento de dados, ao lidar com imagens é necessário levar em conta a posição da observação. Em outras palavras, enquanto no problema de agrupamento de dados seria natural trabalhar apenas com a observação $f(i, j)$, ao realizar segmentação de imagens será necessário considerar toda a informação presente no pixel, isto é, empregar também a posição na qual foi observado o valor $((i, j), f(i, j))$. A indexação dos pixels, isto é, a coordenada espacial de cada pixel será denotada indistintamente na forma explícita “coluna-linha” $((i, j))$ ou pelo ordenamento lexicográfico (k) .

Para exemplificar, considerando a imagem monocromática de tamanho 2×2 exibida na Figura 6 e considere o problema de encontrar os grupos induzidos pelas cores preta e branca. Perante a percepção humana, estes grupos são bem definidos quando os pixels de cor preta são associados a um único grupo e os de cor branca a outro. A Tabela 10 (página 48) ilustra a associação da notação atribuída ao conjunto de dados e a imagem de exemplo. O resultado ideal para este agrupamento segundo a percepção humana é a partição $\mathcal{P} = \{\Omega_1, \Omega_2\}$, considerando $\Omega_1 = \{1, 2\}$ e $\Omega_2 = \{3, 4\}$, onde os indivíduos (pixels) estão representados pelo ordenamento lexicográfico (k) .

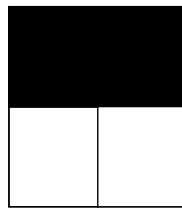


Figura 6: Imagem monocromática.

Muitos algoritmos de agrupamento recebem como entrada uma matriz $n \times n$ contendo as medidas de dissimilaridade entre o conjunto de dados, ou seja, entre seus indivíduos (pixels). Assim, para a Figura 6, tem-se a matriz de dissimilaridade mostrada na Tabela 11 (página 48), considerando a medida de dissimilaridade a distância Euclidiana.

No exemplo anterior, o vetor de atributos é univariado, ou seja só possui uma variável (atributo) escalar para cada indivíduo. Quando $b > 1$ cada posição possui um vetor de atributos multivariado, e estamos diante de imagens multi- ou hiperespectrais. Para exemplificar estes tipos de imagens, consideremos a imagem colorida de

tamanho 3×3 da Figura 7. Ela é composta por três matrizes que representam cada componente do sistema RGB e definem o vetor de atributos de cada pixel com três variáveis, ou seja, $x_k = \{x_{k1}, x_{k2}, x_{k3}\}$. A Tabela 12 (página 48) apresenta o conjunto de dados necessários ao agrupamento, tendo uma matriz de dissimilaridade com ordem 9×9 ilustrada na Tabela 13 (página 49). O resultado ideal para este conjunto de dados é um conjunto partição formado por $\mathcal{P} = \{\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5\}$, considerando $\Omega_1 = \{1, 4\}$, $\Omega_2 = \{7\}$, $\Omega_3 = \{2, 8\}$, $\Omega_4 = \{5\}$, $\Omega_5 = \{3, 6, 9\}$.

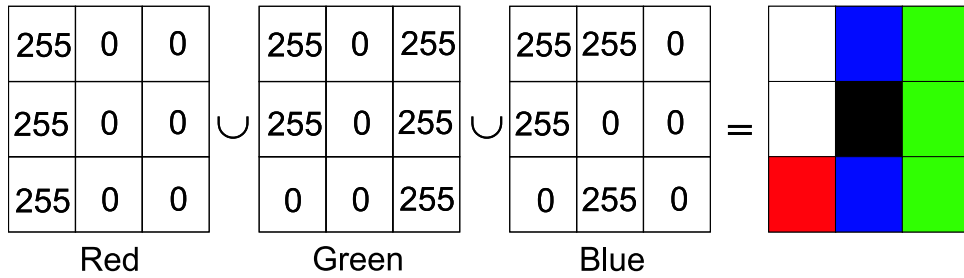


Figura 7: Imagem colorida.

O próximo capítulo descreve o método utilizado nesta dissertação. Ele realiza o agrupamento de dados de forma hierárquica, porém diferencia-se dos algoritmos tradicionais por fazer uma analogia entre o problema de agrupamento e a procura de configurações típicas de um modelo físico: o modelo de Potts (Wiseman, Blatt & Domany 1998). Além disso, serão descritos a aplicação do método à segmentação de imagens e o programa desenvolvido.

ENVI
description = {Registration Result. Method: 1st degree Polynomial w/nearest neighbor}
samples = 709
lines = 946
bands = 3
header offset = 0
file type = ENVI Standard
data type = 1
interleave = bsq
sensor type = Landsat TM
byte order = 0
map info = {UTM, 1, 1, 295380.000, 4763640.000, 30.000000, 30.000000, 13, North}
z plot range = {0.00, 255.00}
z plot titles = {Wavelength, Reflectance}
band names = {Band1, Band2, Band3}
wavelength = {0.485000, 0.560000, 0.660000}

Tabela 1: Exemplo de um arquivo .HDR.

ω_i/ω_j	1	2	3	4	5
1	0				
2	2	0			
3	6	5	0		
4	10	9	4	0	
5	9	8	5	3	0

Tabela 2: Matriz de dissimilaridade D_1 .

	{1,2}	3	4	5
{1,2}	0			
3	5	0		
4	9	4	0	
5	8	5	3	0

Tabela 3: Matriz de dissimilaridade D_2 .

	{1,2}	3	{4,5}
{1,2}	0		
3	5	0	
{4,5}	8	4	0

Tabela 4: Matriz de dissimilaridade D_3 .

ω_i	x_1	x_2
1	1	1
2	1	2
3	6	3
4	8	2
5	8	0

Tabela 5: Exemplificação de um dado de entrada.

ω_i/ω_j	1	2	3	4	5
1	0.00				
2	1.00	0.00			
3	5.39	5.10	0.00		
4	7.07	7.00	2.24	0.00	
5	7.07	7.28	3.61	2.00	0.00

Tabela 6: Matriz de dissimilaridade C_1 .

	{1,2}	3	4	5
{1,2}	0.00			
3	5.22	0.00		
4	7.02	2.24	0.00	
5	7.16	3.61	2.00	0.00

Tabela 7: Matriz de dissimilaridade C_2 .

	{1,2}	3	{4,5}
{1,2}	0.00		
3	5.22	0.00	
{4,5}	7.02	2.83	0.00

Tabela 8: Matriz de dissimilaridade C_3 .

Objetos/Variáveis	1	2	3	4
A	0	0	1	1
B	1	1	0	0
C	1	1	0	0
D	1	1	1	0

Tabela 9: Exemplo de conjunto de dados com variáveis binárias.

k	(i, j)	x_k
1	(0, 0)	0
2	(1, 0)	0
3	(0, 1)	255
4	(1, 1)	255

Tabela 10: Conjunto de dados da imagem monocromática.

k/k	1	2	3	4
1	0	0	255	255
2	0	0	255	255
3	255	255	0	0
4	255	255	0	0

Tabela 11: Matriz de dissimilaridade da imagem monocromática.

k	(i, j)	x_{k1}	x_{k2}	x_{k3}
1	(0, 0)	255	255	255
2	(1, 0)	0	0	255
3	(2, 0)	0	255	0
4	(0, 1)	255	255	255
5	(1, 1)	0	0	0
6	(2, 1)	0	255	0
7	(0, 2)	255	0	0
8	(1, 2)	0	0	255
9	(2, 2)	0	255	0

Tabela 12: Conjunto de dados da imagem colorida.

k/k	1	2	3	4	5	6	7	8	9
1	0	361	361	0	442	361	361	361	361
2	361	0	361	361	255	361	361	0	361
3	361	361	0	361	255	0	361	361	0
4	0	361	361	0	442	361	361	361	361
5	442	255	255	442	0	255	255	255	255
6	361	361	0	361	255	0	361	361	0
7	361	361	361	361	255	361	0	361	361
8	361	0	361	361	255	361	361	0	361
9	361	361	0	361	255	0	361	361	0

Tabela 13: Matriz de dissimilaridade da imagem colorida.

3 Agrupamento Superparamagnético

Neste capítulo será descrito o funcionamento do método de agrupamento superparamagnético, partindo-se do problema de agrupamento de dados, incluindo o modelo físico utilizado (o modelo de Potts não homogêneo), o método de *Swendsen-Wang* para a simulação de ocorrências deste modelo, a localização das regiões superparamagnéticas e, por fim, a identificação dos clusters de dados. Além disso, será exemplificado sua aplicação a imagens e explicado o programa SPC desenvolvido pelo professor Eytan Domany.

Tal como descrito no Capítulo 2, o problema de agrupar dados envolve alguns critérios: indivíduos semelhantes entre si são classificados como pertencentes ao mesmo grupo; indivíduos de grupos distintos são diferentes entre si; todo indivíduo deve pertencer a algum grupo e cada indivíduo faz parte de apenas um único grupo.

Recentemente foi estabelecida uma analogia entre o problema de agrupamento e a procura de configurações típicas de um modelo físico, o modelo de Potts (Wiseman et al. 1998). Dado que existem algoritmos eficientes para a localização dessas configurações, é possível aplicar essas técnicas para uma grande diversidade de situações. As seções seguintes descrevem o método e essa analogia, visando a sua aplicação ao problema de segmentação de imagens.

3.1 Definição

O método de agrupamento por analogia com o comportamento superparamagnético do modelo de Potts é uma técnica para agrupamento de dados baseada nas propriedades físicas de um sistema que exhibe comportamento magnético em função de um parâmetro conhecido como “temperatura”. Esta técnica foi proposta por Blatt, Wiseman & Domany (1996), e posteriormente refinada e aplicada a diversos domínios (ver, por exemplo Blatt, Wiseman & Domany 1997, Wiseman et al. 1998, Domany et al. 1999). Em Ferber & Wörgötter (2000), Neirrotti, Kurcbart & Caticha (2003) e Peng, Urbanc, Cruz, Hyman & Stanley (2003), o algoritmo superparamagnético é aplicado à segmentação de imagens.

O termo “superparamagnético” faz referência ao fato deste modelo exibir três tipos de comportamento associados a outras tantas “regiões” da temperatura: um ferro-

magnético (a temperaturas baixas), um paramagnético (a temperaturas altas) e um superparamagnético. Esta última, que ocorre a temperaturas intermediárias entre altas e baixas, é capaz de fornecer evidência para fazer o agrupamento de dados.

As próximas seções fazem um detalhamento do modelo de Ising (Seção 3.2), o mais simples para entender o comportamento magnético de sistemas não triviais de partículas, para depois generalizá-lo para o modelo de Potts (Seção 3.3), que é a base do agrupamento superparamagnético. Uma vez definido o modelo de Potts, será descrito o algoritmo de Swendsen-Wang (Seção 3.4) que permite gerar eficientemente ocorrências deste modelo e, portanto, é central para a implementação da técnica de agrupamento superparamagnético.

3.2 O Modelo de Ising

A proposta inovadora para resolver o problema de agrupamento pelo método superparamagnético é utilizar o modelo de Potts, que é uma generalização do modelo Ising (ver, por exemplo, Bustos et al. 1998, Pickard 1987, Wu 1982). Nesta seção serão apresentadas as principais propriedades deste último modelo.

Consideremos uma grade bi-dimensional $S = \{-M, \dots, M\} \times \{-N, \dots, N\}$, onde existe um átomo em cada um dos $(2M + 1)(2N + 1)$ pontos $s \in S$ da grade (ver Figura 8, página 51). No que segue será tratado o caso finito, mas as propriedades interessantes do modelo emergem ao considerar o caso infinito, isto é, quando $M, N \rightarrow \infty$.

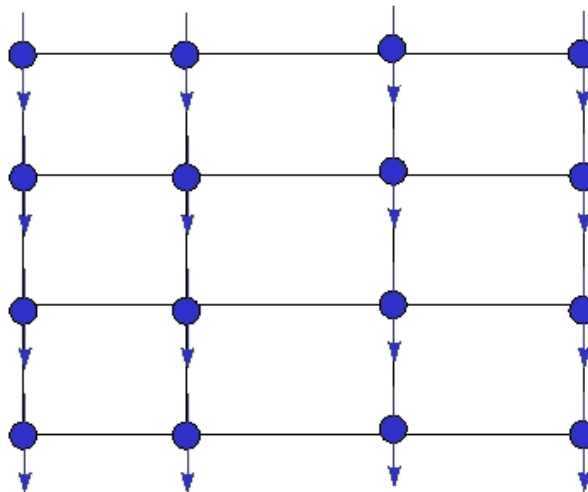


Figura 8: Grade Bidimensional.

O átomo associado a cada coordenada $s \in S$ da grade pode adotar um entre dois estados possíveis (*spin*), estados esses que podem ser entendidos como representativos da orientação do campo magnético criado pela rotação de uma partícula carregada eletricamente; podemos imaginar elétrons em cada coordenada s da grade

S . Dado que essa orientação flutua entre os dois estados possíveis sem que o observador tenha controle sobre ela, a mesma pode ser convenientemente descrita por uma variável aleatória $X_s: \Omega \rightarrow \{-1, +1\}$ (algumas representações utilizam $\{0, 1\}$ como espaço de estados, sem consequência para o resto da discussão). O conjunto de todos os estados adotados pelas variáveis aleatórias $(X_s)_{s \in S}$ é chamado “configuração” ou “estado”; é usual a notação $\xi = (x_1, \dots, x_{(2M+1)(2N+1)})$ para simbolizar um estado. O conjunto de todos os estados possíveis costuma ser denotado Ξ , e é imediato que ele possui $2^{(2M+1)(2N+1)}$ elementos.

O modelo de Ising consiste em associar uma medida de energia a cada configuração $\xi \in \Xi$. A natureza tenderá a escolher configurações de energia baixa em detrimento de configurações de energia alta.

A energia total do sistema no estado ξ pode ser escrita da seguinte forma:

$$E(\xi) = -J \sum_{\langle u, v \rangle} x_u x_v - H \sum_{s \in S} x_s. \quad (3.1)$$

Na Equação (3.1), x_s depende do sentido do *spin* do átomo, se este *aponta* para cima x_s é igual a $+1$, caso contrário a -1 . A constante $J \in \mathbb{R}$ parametriza o acoplamento entre os *spins* e $H \in \mathbb{R}$ é um campo magnético externo. O primeiro termo da Equação (3.1) é um somatório responsável pela contabilização das contribuições associadas à interação entre uma dada partícula e suas vizinhas. Para os propósitos do nosso trabalho, poderemos considerar $H = 0$ mas as interações J irão variar conforme as partículas que interagem. A notação “ $\langle u, v \rangle$ ” indica que devem ser somados todos os pares de vizinhos $u, v \in S$.

Caso sejam consideradas interações de primeira ordem, um átomo que esteja na posição $u = (i, j)$ da rede pode interagir apenas com seus vizinhos nas posições $(i+1, j)$, $(i-1, j)$, $(i, j+1)$, $(i, j-1)$ (esta é chamada também interação de vizinhança-4).

O primeiro termo da equação faz com que a energia do sistema seja mínima (máxima) quando os *spins* estiverem alinhados paralelamente (antiparalelamente) no caso em que J é positivo, ou seja, J positivo favorece o **ferromagnetismo**.

No nosso caso o campo externo poderá ser considerado nulo, isto é, $H = 0$. Tal como apresentado pela Equação (3.1), onde as interações entre partículas são descritas por uma única constante J , temos o chamado “modelo de Ising homogêneo”. Para as aplicações de interesse neste trabalho, as interações irão depender das posições e, com isso, o modelo de Ising (agora não homogêneo) estará caracterizado pela energia

$$E(\xi) = - \sum_{\langle u, v \rangle} J(u, v) x_u x_v. \quad (3.2)$$

Um parâmetro muito conveniente para descrever o comportamento do modelo de

Ising é a temperatura, que aparece escrevendo a energia da configuração ξ como

$$E_T(\xi) = -\frac{1}{T} \sum_{\langle u,v \rangle} J(u,v)x_u x_v, \quad (3.3)$$

com $T > 0$.

Será conveniente no decorrer do texto empregar uma formulação equivalente à utilizada na Equação (3.3): ao invés de somar produtos de *spins* vizinhos e, com isso, termos um somatório de “+1” e “-1”, podemos mapear este somatório para um outro de somandos “1” e “0”, respectivamente. Para isto, a Equação (3.3) pode ser escrita

$$E_T(\xi) = -\frac{1}{T} \sum_{\langle u,v \rangle} J(u,v)\mathbb{I}(x_u, x_v), \quad (3.4)$$

onde a função $\mathbb{I}(a,b)$ vale 1 se $a = b$ e zero caso contrário. A diferença entre as formulações dada pelas Equações (3.2) e (3.4) é uma constante na energia da configuração, constante essa que não terá influência nenhuma no modelo.

O modelo de Ising é um modelo estocástico, pois ele associa uma probabilidade a cada configuração $\xi \in \Xi$:

$$\Pr_T(\xi) = \frac{1}{Z_T} \exp\{-E_T(\xi)\},$$

onde $Z_T = \sum_{\xi \in \Xi} \exp\{-E_T(\xi)\}$ é chamada “constante de partição” ou ainda “constante de normalização”.

Na próxima seção será detalhado o modelo de Potts com base nas definições apresentadas nesta seção.

3.3 O Modelo de Potts

Uma generalização para o modelo de Ising, que é limitado a apenas dois estados, é o modelo de Potts, onde cada elemento pode assumir um de k estados possíveis $\{\zeta_1, \dots, \zeta_k\}$. Este modelo descreve o comportamento de partículas que interagem entre si a uma temperatura $T > 0$. Sejam as n partículas $\mathbf{x} = \{x_{i,j}\}_{i,j=1}^n$ interagindo todas entre si. A partícula x_u interage (atrativamente) com a partícula x_v com intensidade $\beta(u,v) \geq 0$. A distribuição da configuração ζ é dada por

$$\Pr_T(\mathbf{x} = \zeta) = \frac{1}{Z_T} \exp\left\{\frac{1}{T} \sum_{u,v} \beta(u,v)\mathbb{I}_{u,v}\right\} \quad (3.5)$$

onde

$$\mathbb{I}_{u,v} = \begin{cases} 1 & \text{se } \zeta_u = \zeta_v \\ 0 & \text{caso contrário,} \end{cases}$$

onde $Z_T = \sum_{\zeta \in \Xi} \exp\{T^{-1} \sum_{u,v} \beta(u,v)\mathbb{I}_{u,v}\}$ é uma constante de normalização e Ξ é o conjunto de todas as configurações possíveis. Esta distribuição sempre irá favorecer

às configurações de partículas do mesmo tipo, mais intensamente quanto menor for a temperatura.

No modelo clássico de Ising, temos que as interações consideradas dependem apenas da distância entre as partículas, e estas estão localizadas sobre uma grade regular. Já no modelo de Potts não homogêneo, que é um caso mais geral aqui considerado, a localização das partículas não é necessariamente levada em conta, pois todas podem interagir entre si.

3.4 O Método de Swendsen-Wang

O método de Swendsen-Wang é uma dinâmica que permite simular ocorrências do modelo de Potts de forma iterativa. Essas simulações são largamente usadas para estimar a esperança de variáveis aleatórias de um sistema de Potts, dado que freqüentemente é impossível avaliá-las de forma analítica.

Simulação de Monte Carlo é largamente utilizada em estudos de mecânica estatística, mas especificamente em modelos do magnetismo baseados em *spins*. Algoritmos de Monte Carlo tradicionais (Metropolis & Ulam 1949) para estes modelos, tal como o algoritmo de Metropolis (Metropolis, Rosebluth, Rosebluth, Teller & Teller 1953), apresentam um decremento rápido na eficiência computacional em função do aumento do tamanho do sistema. O algoritmo desenvolvido por Swendsen e Wang (Swendsen 1991, Swendsen & Wang 1987) realiza mudanças globais (não locais) nas configurações de *spins*, diferentemente do algoritmo de Metropolis, que é um algoritmo local. O algoritmo de Swendsen-Wang reduz muito o decremento da eficiência computacional para certos tipos de modelo.

Este método é uma das etapas do algoritmo de agrupamento de dados (descrito na próxima seção) e seu algoritmo está descrito nos passos seguintes.

1. Considere uma configuração qualquer dos *spins*, ou seja, a primeira configuração é arbitrária. Em particular, e sem perda de generalidade, considere a configuração onde cada *spin* é gerado independentemente dos outros com probabilidade uniforme nos k estados possíveis $\{\zeta_1, \dots, \zeta_k\}$. Esta situação é exibida na Figura 9 (página 55), onde são considerados $n = 10$ indivíduos e $k = 3$ estados possíveis.
2. Em seguida, todos os pares de *spins* $\langle u, v \rangle$ que interagem devem ser visitados. Se os dois *spins* estiverem no mesmo estado, isto é se $\zeta_u = \zeta_v$, então é calculada a probabilidade de ser estabelecido um **arco** (ou ligação) entre esses dois *spins*, que é dada pela Equação (3.5) (página 53).
3. Após o cálculo da probabilidade do passo anterior é observada $u = U(\omega)$, ocorrência de uma variável aleatória com distribuição uniforme no intervalo $(0, 1)$, isto é, $U \sim \mathcal{U}(0, 1)$. Se $u < \Pr_T(\mathbf{x} = \zeta)$ então um arco é estabelecido entre os dois *spins*. Caso contrário, nenhuma ligação é estabelecida.

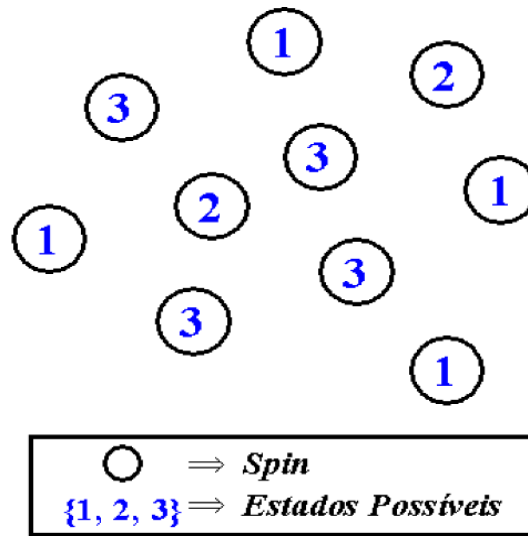


Figura 9: Associação entre spins e estados

Para melhor ilustrar os passos 2 e 3 considere o exemplo da Figura 9. Se inicialmente e ao acaso escolhermos o *spin* destacado da Figura 10 para iniciar o passo 2, teremos os *spins* destacados da Figura 11 (página 56) que satisfazem a condição $\zeta_u = \zeta_v$, ou seja, apresentam-se no mesmo estado.

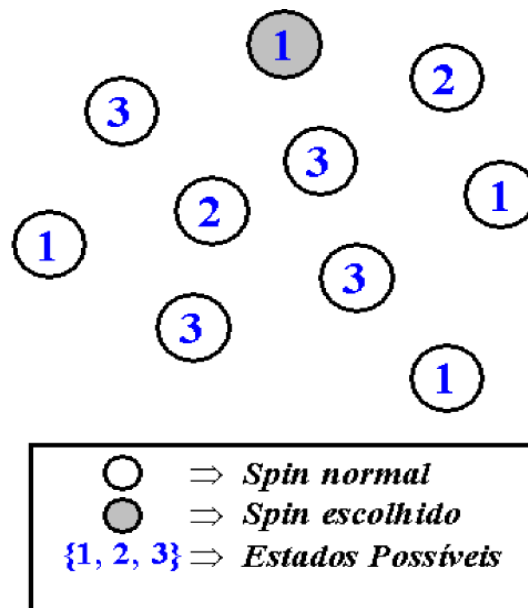


Figura 10: Escolha do *spin* inicial

Calcula-se a probabilidade $\Pr_T(\mathbf{x} = \zeta)$ e são observadas ocorrências independentes da variável aleatória $U \sim U(0, 1)$, uma para cada par de partículas. Um arco é estabelecido entre os pares de *spins* onde a ocorrência u for menor do que a probabilidade $\Pr_T(\mathbf{x} = \zeta)$. Supondo que a probabilidade obtida para o par de *spins* destacado na Figura 12 (página 57) seja 0.8 e que, aleatoriamente, o valor de u obtido seja 0.6, de acordo com o passo 3 será estabelecida uma ligação entre o par.

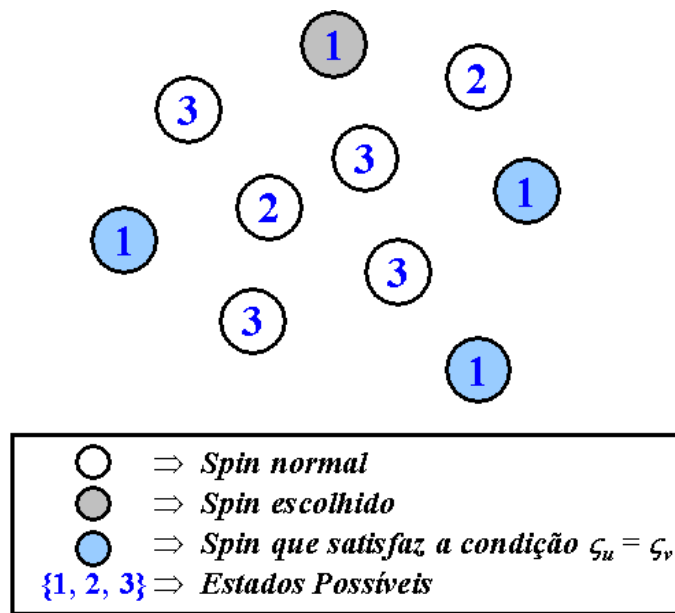


Figura 11: *Spins* que satisfazem a condição $\zeta_u = \zeta_v$

4. Uma vez visitados todos os pares de *spins* e estabelecidas todas as associações onde $u < \Pr_T(x = \zeta)$, o próximo passo é identificar os grupos de *spins* de Swendsen-Wang (SW). Um grupo SW contém todos os *spins* que possuem uma ligação entre eles. De acordo com a Equação (3.5) pode-se notar que somente os *spins* com os mesmos valores, ou seja, que estão no mesmo estado, poderão fazer parte de um mesmo grupo SW. Uma possível configuração e a representação de grupos de Swendsen-Wang podem ser vistas na Figura 13 (página 58).
5. A etapa final do procedimento é gerar uma nova configuração a partir da anterior através da escolha aleatória de um dos k estados possíveis para cada grupo de Swendsen-Wang.

A Figura 14 (página 59) representa este último passo. Nela pode ser observado que um dos *spins* que estava no estado 1 e que fazia parte de um grupo SW, contendo outros dois *spins*, foi escolhido e o estado de todos que pertenciam a esse grupo foi substituído pelo novo estado escolhido aleatoriamente, no caso do exemplo o estado escolhido foi o 2.

6. Retorna-se ao passo 2 enquanto não for satisfeito algum critério de convergência.

Os trabalhos citados anteriormente (Swendsen 1991, Swendsen & Wang 1987) mostram que, seja qual for a configuração inicial e para toda temperatura T , após um certo número de iterações a configuração observada pode ser considerada oriunda do modelo de Potts.

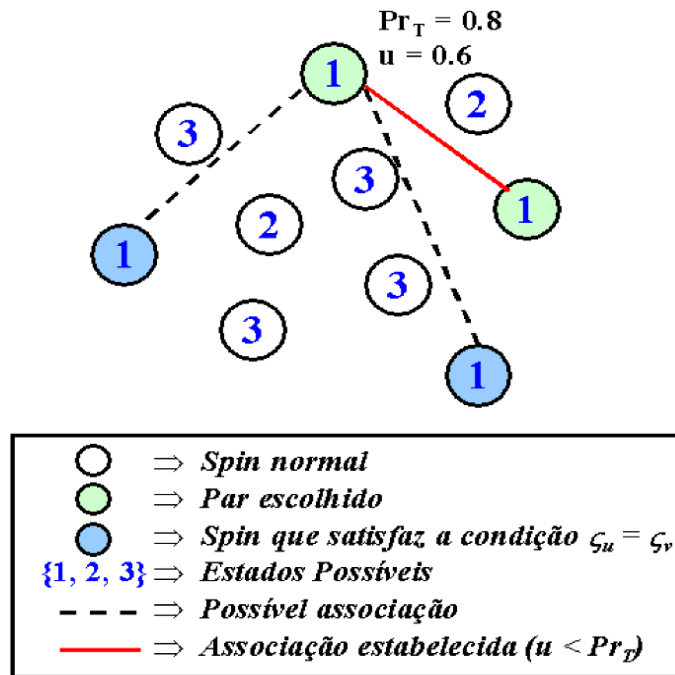


Figura 12: Estabelecimento de arco entre spins - Passo 3

3.5 Visão Geral do Algoritmo de Agrupamento de Dados

O método de agrupamento de dados a ser descrito nesta seção é baseado no modelo de Potts não homogêneo (ver Seção 3.3) e no método de Swendsen-Wang abordado na Seção 3.4.

Considere que o conjunto de dados consiste de N indivíduos, cujas dissimilaridades são denotadas por $d_{i,j}$. Será observada a evolução do modelo de Potts formado por N partículas (cada uma representando um dos N indivíduos). As partículas i e j interagem com uma força $J_{i,j}$ que é inversamente proporcional à dissimilaridade $d_{i,j}$. Reescrevendo a Equação (3.5, página 53), o Hamiltoniano (a energia) do modelo de Potts assim definido é dado por

$$Pr_T(\mathbf{x} = \zeta) = \frac{1}{Z_T} \exp\left\{\frac{1}{T} \sum_{u,v} J_{i,j} \mathbb{I}_{u,v}\right\}. \quad (3.6)$$

Dado que as interações $J_{i,j}$ são todas não negativas, a configuração mais provável, isto é, de menor energia é aquela que atribui o mesmo estado a todas as partículas. Analogamente, a configuração menos provável é aquela que atribui um estado diferente para cada partícula.

A temperaturas muito baixas, o modelo de Potts ficará muito freqüentemente “congelado”, isto é, com a maioria das partículas no mesmo estado. A temperaturas muito altas, o mesmo modelo freqüentemente exibirá estados onde a maioria das partículas adota estados diferentes.

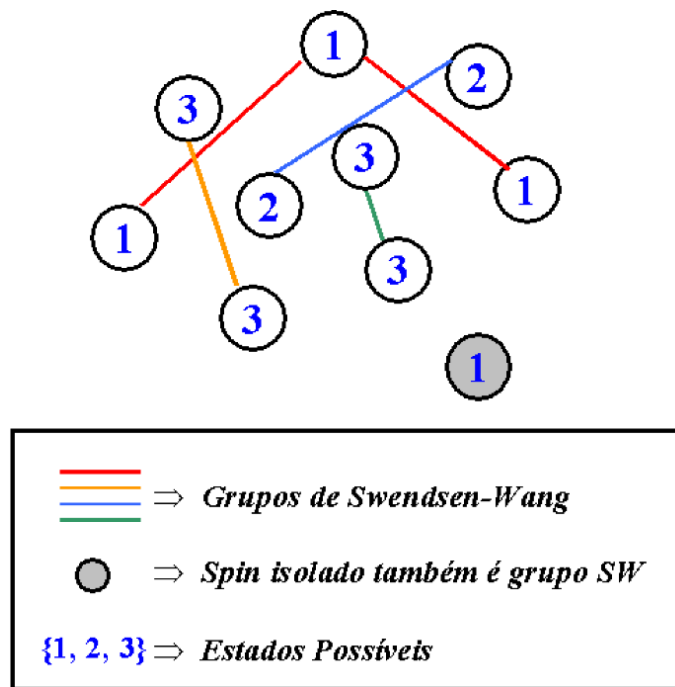


Figura 13: Configuração com grupos de SW

Em termos de agrupamento, a primeira situação é aquela onde todos os indivíduos foram associados no mesmo grupo, e a última aquela onde cada indivíduo forma um grupo diferente dos outros. Nenhum desses dois extremos é interessante na maioria das aplicações práticas.

A proposta de Blatt et al. (1996), refinada nos artigos posteriores (Blatt et al. 1997, Wiseman et al. 1998, Domany et al. 1999), consiste em adotar uma seqüência de temperaturas entre uma temperatura muito baixa e uma temperatura muito alta e, para cada uma dessas temperaturas, observar uma seqüência de ocorrências do modelo de Potts para estimar a correlação entre partículas. Duas observações serão consideradas no mesmo grupo se a correlação entre os respectivos estados for superior a um limiar previamente estabelecido.

A dinâmica de Swendsen-Wang é empregada para gerar essas seqüências de ocorrências do modelo de Potts. A vantagem de empregar esta dinâmica em relação a usar as dinâmicas de Metropolis (Metropolis & Ulam 1949, Metropolis et al. 1953) ou o *Gibbs sampler* (Geman & Geman 1984) reside nas seguintes propriedades:

1. esta dinâmica converge rapidamente seja qual for a configuração inicial;
2. uma vez atingida a convergência, são necessários poucas iterações entre duas configurações para que as mesmas possam ser consideradas independentes (a independência é essencial para poder fazer estimativas confiáveis de, por exemplo, a correlação entre estados);
3. é possível implementá-la de forma eficiente com linguagens de programação

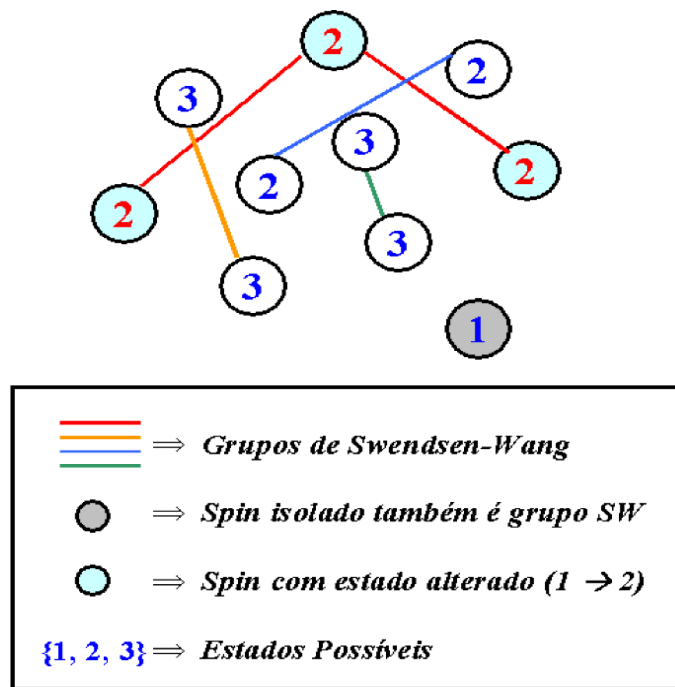


Figura 14: Nova configuração gerada

usuais.

As etapas do algoritmo são detalhadas nos passos a seguir:

1. Realizar a analogia física com o problema de spins de Potts
 - (a) Associar a cada indivíduo ω_i um *spin* de Potts caracterizado por um de q estados possíveis $\zeta_i = 1, 2, \dots, q$.
 - (b) Identificar os vizinhos de cada ponto ω_i de acordo com um critério pré-selecionado; no problema de agrupamento de dados todos os indivíduos são potencialmente vizinhos.
 - (c) Calcular a interação J_{ij} entre os vizinhos ω_i e ω_j de forma que seja inversamente proporcional à dissimilaridade entre esses indivíduos.
2. Localizar a fase superparamagnética, ou para vários valores da temperatura entre uma temperatura baixa e uma temperatura alta
 - (a) Estimar a magnetização média para diferentes temperaturas.
 - (b) Usar a susceptibilidade para identificar a fase superparamagnética.
3. Determinar dentro da fase superparamagnética as seguintes medidas:
 - (a) Medir a correlação spin-spin para todos os pontos vizinhos ω_i, ω_j .
 - (b) Construir os grupos de dados.

As seções seguintes detalham as etapas do algoritmo de agrupamento.

3.5.1 Analogia física com o problema de spins de Potts

O objetivo desta etapa é especificar o Hamiltoniano na forma da Equação (3.6, página 57), que serve como a analogia física dos pontos de dados a serem agrupados.

Após definir q , o número de estados possíveis que um *spin* de Potts pode assumir, é atribuído um *spin* de Potts a cada ponto de dado escolhendo um dentre os q valores possíveis com probabilidade q^{-1} independentemente das outras posições.

Outra característica importante a ser definida é o conceito de “vizinhança”, bastante utilizado na execução do algoritmo. No caso de segmentação de imagens só será considerada interação entre posições próximas espacialmente; já para agrupamento de dados todas as posições são potencialmente vizinhas entre si. Por fim é fornecida a dependência funcional da força da interação J_{ij} na distância entre os *spins* vizinhos.

Esta seção discute as possíveis escolhas para esses atributos do Hamiltoniano e suas influências na performance do algoritmo. A observação mais importante a ser feita é que o algoritmo é suficientemente robusto para produzir resultados bons para uma larga classe de especificações. Em outras palavras, não há uma especificação “ótima” em detrimento de outras especificações. O algoritmo irá produzir bons resultados sempre que uma escolha razoável for feita, e a faixa de escolhas razoáveis é muito ampla.

Associação dos Spins de Potts a cada ponto ω_i

O número de estados de Potts q determina principalmente a sutileza das transições e as temperaturas nas quais elas ocorrem. Por outro lado, para manter uma dada precisão estatística é necessário executar simulações muito longas à medida que o valor de q cresce. A partir de testes realizados por Domany et al. (1999), e verificados no nosso trabalho, foi possível concluir que a influência de q na classificação resultante é fraca, e por isso ao longo do projeto foi atribuído o valor $q = 20$ para todos os exemplos.

Uma vez escolhido o número de estados a ser utilizado no modelo de Potts, o passo seguinte é associar a cada *spin* um possível estado, como pode ser visto na Figura 15 (página 61).

Identificação dos vizinhos

A necessidade para a identificação dos vizinhos do ponto ω_i poderia ser eliminada se todos os pares i, j de *spins* de Potts interagissem com cada outro via uma interação de curta distância $J_{ij} = f(d_{ij})$, que decresce exponencialmente com a distância entre os dois pontos de dados. As fases e as propriedades do modelo não serão fortemente afetadas pela escolha da forma precisa de f . Computacionalmente esse decrescimento exponencial exigiria um grande gasto, então foi verificado que esse problema seria

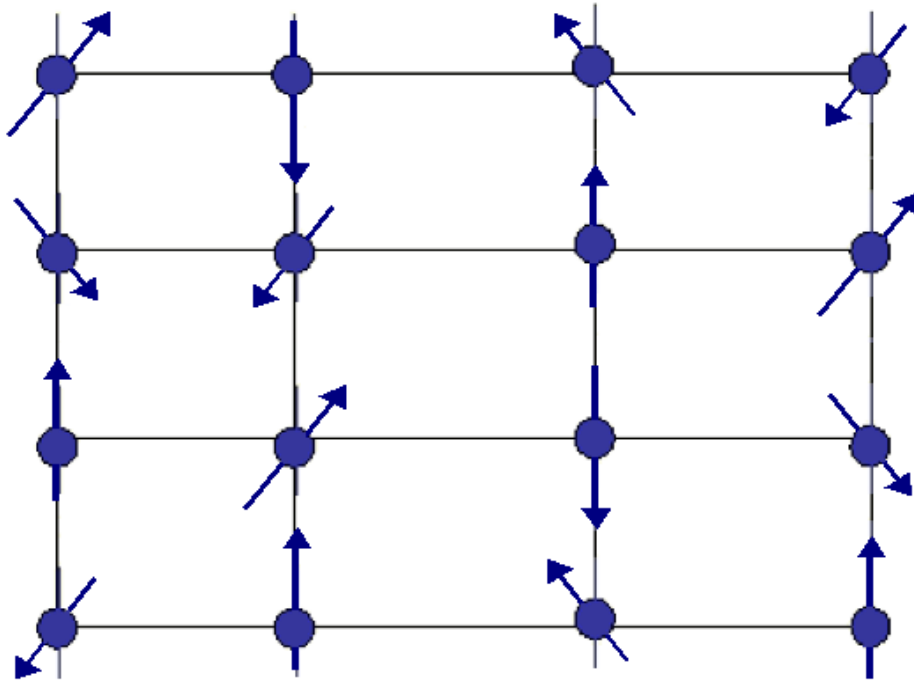


Figura 15: Associação de estados do modelo de Potts a *Spins*.

resolvido se fossem mantidas somente as interações de um *spin* com um número limitado de vizinhos, e atribuído a todas as outras J_{ij} igual a zero. Uma interação $J_{ij} = 0$ é equivalente a afirmar que a similaridade entre as observações i e j é nula, ou que a sua dissimilaridade é infinita; neste caso essas observações jamais serão agrupadas, a não ser que todas as observações sejam colocadas na mesma classe.

Uma vez que os dados não formam uma grade regular, é preciso fornecer alguma definição razoável para “vizinhos”. Sendo assim, para dimensões pequenas, onde $d \leq 3$, foram utilizadas características da triangularização sobre estruturas de grafos em agrupamentos de dados (Ahuja 1982). Os conjuntos de dados usados nos nossos exemplos possuem grandes dimensões, logo, maior atenção é dada a dimensões onde $d > 3$.

Para grandes dimensões, é usado o valor de vizinhança mútua, onde ω_i e ω_j têm um valor de vizinhança mútua K , se e somente se ω_i é um dos K vizinhos mais próximos de ω_j e ω_j é um dos K vizinhos para mais próximos de ω_i . O valor de K foi escolhido de tal maneira que as interações conectem todos os pontos de dados a um grafo conectado. Claramente K cresce com a dimensão. Por motivos computacionais, tornou-se conveniente, em casos onde a dimensão é muito alta ($d > 100$, por exemplo) fixar $K = 10$.

Cálculo das interações locais

Para ter um modelo com as propriedades físicas de um imã granular não-homogêneo, é preciso obter fortes interações entre *spins* que correspondam a dados de uma região de alta densidade e fracas interações entre *spins* que estejam em regiões de baixa densidade. Para este fim, e em comum com outros métodos locais, será suposto que existe uma quantidade local a , definida pelas regiões de alta densidade e menor do que a distância típica entre pontos nas regiões de baixa densidade. Este valor a é a quantidade característica sobre a qual nossas interações de curta distância decaem. Uma boa escolha para calcular J_{ij} é a seguinte:

$$J_{i,j} = \begin{cases} \frac{1}{\tilde{K}} \exp\left(\frac{-d_{ij}^2}{2a^2}\right) & \text{se } \omega_i \text{ e } \omega_j \text{ forem vizinhos} \\ 0 & \text{caso contrário,} \end{cases} \quad (3.7)$$

onde, segundo Domany et al. (1999), a quantidade a será a média de todas as distâncias d_{ij} entre a vizinhança dos pares ω_i e ω_j , e \tilde{K} é o número médio de vizinhos. Todos os detalhes vistos até agora podem ser facilmente implementados quando ao invés do fornecimento dos vetores de atributos x_i de todos os dados, for obtida uma matriz $N \times N$ de dissimilaridades d_{ij} .

3.5.2 Localização da fase superparamagnética

Os vários intervalos de temperatura nos quais o sistema se auto-organiza em diferentes partições para grupos são identificados através da medida de susceptibilidade χ , que é uma função da temperatura. A susceptibilidade é a variância da magnetização média para cada temperatura, e deve ser estimada a partir das simulações Monte Carlo.

O método de Monte Carlo foi utilizado com o seguinte procedimento, que inclui a dinâmica de Swendsen-Wang:

1. Escolher o número de iterações M a ser executada.
2. Gerar uma configuração inicial através da atribuição de um valor aleatório a cada *spin*.
3. Associar um arco entre os pontos vizinhos ω_i e ω_j com probabilidade Pr_T (ver Equação 3.6 na página 57).
4. Encontrar os subgrafos conectados, os grupos de SW.
5. Atribuir novos valores aleatórios aos *spins* (aos *spins* que pertencem ao mesmo grupo SW são associados o mesmo valor). Isto é a nova configuração do sistema.
6. Calcular o valor assumido pelas quantidades físicas de interesse na nova configuração de *spin*.

7. Voltar para o passo 3 a menos que o número de iterações, M , tenha sido alcançado.
8. Calcular a magnetização média, dada pelo tamanho do maior grupo dividido pelo número total de observações.

A fase superparamagnética pode conter muitos e diferentes subgrafos com propriedades diferentes. A susceptibilidade χ foi medida em diferentes temperaturas para localizar diferentes regimes. O objetivo é identificar as temperaturas nas quais o sistema muda sua estrutura.

Existem duas características básicas da susceptibilidade que possuem uma grande importância no trabalho. A primeira é um pico na susceptibilidade, que sinaliza uma transição da fase ferromagnética para superparamagnética, na qual um grupo grande é quebrado em pequenos grupos. A segunda característica é um decréscimo abrupto da susceptibilidade, correspondendo a uma transição da fase superparamagnética para a paramagnética, na qual um ou mais grupos grandes são quebrados em muitos grupos pequenos.

3.5.3 Determinação de medidas na fase superparamagnética

Uma vez que a fase superparamagnética e suas diferentes subfases tenham sido identificadas, é selecionada uma temperatura em cada região de interesse. A lógica é que cada subfase caracteriza um tipo particular de partição dos dados, com novos grupos sendo formados ou sendo divididos. Por outro lado, como a temperatura é variada dentro de uma fase, espera-se somente o encolhimento (retrocesso) ou a expansão dos grupos existentes, mudando somente a classificação dos pontos nas fronteiras (limites) dos grupos.

A Correlação *Spin-Spin*

A função de correlação *spin-spin* G_{ij} entre os indivíduos ω_i e ω_j é usada para construir os grupos de dados. O método Monte Carlo permite calcular estimadores da função de correlação *spin-spin*, em particular pelo fato de empregar a dinâmica de Swendsen-Wang que, como já mencionado, é de rápida execução.

Um estimador simples para a correlação entre os pontos ω_i e ω_j pode ser obtido a partir de um estimador da probabilidade desses indivíduos estarem no mesmo grupo. Este estimador pode ser a média das funções indicadoras calculada sobre M ocorrências (idealmente independentes) do modelo de Potts, isto é, estimaremos G_{ij} calculando $\hat{C}_{ij} = \sum_{\ell=1}^M \mathbb{I}_{i,j}(\ell)/M$, com

$$\mathbb{I}_{i,j} = \begin{cases} 1 & \text{se } \omega_i \text{ e } \omega_j \text{ pertencem ao mesmo grupo} \\ 0 & \text{caso contrário,} \end{cases}$$

que é um estimador da probabilidade de encontrar os indivíduos ω_i e ω_j no mesmo grupo, e finalmente computando

$$\hat{G}_{ij} = \frac{(q-1)\hat{C}_{ij} + 1}{q},$$

que é usado como estimador da correlação G_{ij} .

Os grupos de dados

Os grupos são identificados em três passos:

1. Construir o centro dos grupos usando um procedimento de limiarização; de $G_{ij} > 0.5$, uma ligação é estabelecida entre os pontos de dados vizinhos ω_i e ω_j . O grafo conectado resultante depende fracamente do valor (0.5) usado nesta limiarização, desde que este limiar esteja no intervalo $(q^{-1}, 1 - 2q^{-1})$. A razão para isto é que a distribuição das correlações entre os dois *spins* vizinhos chega ao ponto máximo fortemente nesses dois valores e é muito pequena entre eles.
2. Capturar pontos que estejam situados na periferia dos grupos através da ligação de cada ponto ω_i a seu vizinho ω_j de correlação máxima estimada \hat{G}_{ij} . Pode acontecer que pontos ω_i e ω_j já tenham sido ligados no passo anterior.
3. Os grupos de dados são identificados como os componentes ligados dos grafos obtidos nos passos 1 e 2.

3.6 Aplicação do Algoritmo SPC a Imagens

Diante das explicações realizadas nas seções anteriores deste capítulo, tem-se que a técnica de agrupamento descrita pode ser aplicada a uma grande variedade de dados. Um caso particular de interesse é sua aplicação ao problema de segmentação de imagens.

Considerando os conceitos fundamentais do modelo físico de Potts, tem-se que cada pixel da imagem é associado a um spin. Quando o spin (pixel) central é similar ao seu vizinho, eles tendem ao alinhamento, isto é, a pertencerem à mesma classe. As diferentes temperaturas induzem diferentes relações de similaridade entre grupos de spins e, para cada temperatura, os grupos formados compõem uma imagem segmentada.

O algoritmo superparamagnético pode, então, ser considerado um método hierárquico de agrupamento de dados, onde a cada variação de temperatura um tipo de segmentação é originada na imagem analisada. A temperatura mais baixa formará uma imagem com um único grupo, ou seja, todos os pixels da imagem pertencem a um mesmo segmento; esta situação é denominada “ferromagnética”, por

exibir um forte alinhamento de spins. A medida que a temperatura aumenta, a quantidade de grupos (segmentos) nesta imagem tenderá a aumentar até a desordem total; esta última situação é chamada “paramagnética”. A imagem segmentada desejada apresentar-se-á tipicamente em alguma temperatura da fase superparamagnética, que é intermediária entre as fases ferromagnética (baixas temperaturas, poucos grupos, subsegmentação) e paramagnética (altas temperaturas, muitos grupos, sobresegmentação).

Considerando os passos que definem o algoritmo, tem-se que a técnica de agrupamento superparamagnético concentra-se em três principais etapas: realizar a analogia física com o problema de spins de Potts, localizar a fase superparamagnética e através desta fase construir os grupos de dados. A primeira etapa faz a associação dos dados a serem agrupados com a técnica proposta. Como os dados de interesse são imagens, precisa-se adaptar os dados da imagem à entrada padrão. Além disso, o critério de vizinhança estabelecido nesta etapa pode identificar a aplicação a ser analisada, por exemplo, se será uma classificação sem rotulação ou uma segmentação.

Como visto na Seção 2.7, para aplicar técnicas de agrupamento de dados a uma imagem é necessário considerarmos cada pixel da imagem como um elemento pertencente ao conjunto dos dados de entrada. No agrupamento superparamagnético, cada observação ω_i é um pixel da imagem. Para exemplificar, considere a Figura 7 (página 45) e representando os pixels pelo ordenamento lexicográfico, tem-se um conjunto de dados de entrada formado por nove pontos, ou seja, $\omega_i \in \{1, \dots, 9\}$. Diante da aplicação proposta, o conjunto de dados passa a ser a imagem a ser segmentada, cujos indivíduos são os pixels que compõem a imagem.

Outro fator importante na primeira etapa do algoritmo superparamagnético é o critério selecionado para estabelecer os vizinhos de cada ponto ω_i . Este critério influenciará diretamente na aplicação a ser tomada: classificação ou segmentação, por exemplo. Quando este critério envolve um domínio baseado nas características da imagem, pode-se obter uma classificação, ou seja, os vizinhos considerados são os mais próximos (semelhantes) perante seus atributos na imagem, mesmo que estes sejam distantes espacialmente. Um exemplo de classificação pode ser o da Figura 7, onde os grupos formados pelas cores branca, verde e outros estariam associados às suas classes correspondentes, considerando cada classe pela cor. Quando o critério de vizinhança envolve o domínio espacial, pode-se obter uma segmentação. Uma explicação mais detalhada deste assunto está feita no próximo capítulo.

4 O Domínio da Aplicação

No capítulo anterior, o algoritmo superparamantético foi explicado em detalhes, mas os estudos realizados nesta dissertação se destinam a aplicar este algoritmo à segmentação de imagens, cujas descrições estão definidas neste capítulo. Para isto, inicialmente, fazem-se explicações sobre segmentação de imagens através de seus conceitos e problemas relacionados a este assunto. Grande parte das explicações sobre o conceito de segmentação de imagens foi baseada nos textos de Banon (2000), de Gonzalez & Woods (2000), de Jain (1989) e de Facon (1993).

4.1 Segmentação de Imagens

Alguns psicólogos alemães no início de século introduziram o princípio da segmentação (ver Facon 1993). Eles mostraram que o ser humano, em termos de percepção pela visão, efetua agrupamentos sobre o que ele percebe numa determinada cena com base na proximidade, similaridade e continuidade. Os agrupamentos realizados pelo ser humano podem ser considerados como “unidades significantes”, cuja composição forma a cena em questão. Além dessa constatação, eles postularam a existência de uma componente decisiva para a compreensão da cena, as “unidades significantes”.

Os critérios estabelecidos para agrupar as informações são a base da segmentação. A proximidade visa o agrupamento de informações consideradas próximas espacialmente, ou seja, associa-se uma informação aos seus vizinhos perante a localização espacial. A similaridade determina se duas informações são parecidas segundo alguma característica, por exemplo, cores. A continuidade está relacionada ao conceito de conectividade num grafo, considerando a “unidade significativa” um grafo, deve-se existir pelo menos um caminho entre duas informações presentes neste grafo. Normalmente, faz-se uma associação entre continuidade e o conceito de conectividade em imagens, resumindo-se nas seguintes propriedades:

- dois pixels estão conectados se eles são adjacentes (se são vizinhos) e se seus níveis de cinza satisfazem um critério de similaridade.
- se p e q forem pixels de um subconjunto S de uma imagem, então p está conectado a q em S se existir um caminho de p a q consistindo inteiramente de pixels de S (ver Banon 2000).

Diantes desses conceitos, pode-se entender o objetivo da segmentação que é obter um conjunto de “primitivas ou segmentos significativos” de uma imagem digital. O nível até o qual estes segmentos são subdivididos depende do problema a ser resolvido. Por exemplo, para comprimir um vídeo com uma bola se movimentando num fundo uniforme, é necessário separar a bola do fundo para só ser analisado o movimento da bola, já que o fundo é desprezível por se repetir. Mas, se ao invés da bola fosse uma pessoa fazendo exercícios com movimentos bruscos nos membros superiores, poderia haver a necessidade de se obter a segmentação membros superiores, corpo sem membros superiores e fundo, e não mais somente a separação entre pessoa e fundo. Esta segmentação possui um nível maior de profundidade com relação ao exemplo da bola.

Os algoritmos de segmentação para imagens monocromáticas são geralmente baseados em duas propriedades básicas associadas aos valores de níveis de cinza: descontinuidade e similaridade. Na primeira categoria, a abordagem é particionar a imagem baseado-se em mudanças abruptas nos níveis de cinza. Quando esta partição é realizada, o resultado é formado pelos contornos dos segmentos. As principais áreas de interesse nessa categoria são a detecção de pontos isolados e detecção de linhas e bordas da imagem. A segunda categoria segmenta a imagem de acordo com a similaridade atribuída em alguma característica da imagem, por exemplo, intensidade. Algumas dessas técnicas tentam encontrar regiões consideradas homogêneas, ou seja, regiões que apresentam alguma propriedade local aproximadamente constante, no caso a similaridade. Estas regiões devem ser disjuntas entre si e a união delas compor a imagem. As principais abordagens da segunda categoria baseiam-se em limiarização, crescimento de regiões e agrupamento de dados. O resumo de algumas dessas técnicas está descrito no Capítulo 1.

Como o modelo mais geral associado à operação de segmentação é aquele no qual a imagem consiste de regiões que obedecem a um certo critério de similaridade, a próxima subseção define este conceito e suas propriedades.

4.1.1 Definição

Uma região de uma imagem é um conjunto de pixels que apresenta duas propriedades:

1. seus pixels devem estar conectados, isto é, de qualquer pixel da região pode-se chegar a qualquer outro pixel por um caminho inteiramente contido nessa região;
2. ela deve ser homogênea, ou seja, ela deve apresentar alguma propriedade local aproximadamente constante como, por exemplo, a similaridade de alguma característica da imagem.

Segundo Haralick & Shapiro (1985), as características desejáveis para se exibir uma imagem segmentada robusta, ou seja, considerada de boa qualidade com base em imagens monocromáticas são:

- as regiões encontradas na imagem segmentada devem ser uniformes e homogêneas segundo algum critério estabelecido, tal como textura uniforme, intensidade média e outros;
- os interiores das regiões devem ser simples e com poucos “buracos”;
- as regiões adjacentes devem possuir diferenças significativas segundo o critério estabelecido;
- as bordas de cada segmento devem ser simples e espacialmente bem definidas.

Diante dessas características, pode-se definir segmentação de imagens como sendo uma partição da imagem em regiões disjuntas e homogêneas segundo algum critério estabelecido. Em Gonzalez & Woods (2000), tem-se uma definição formal: seja S o suporte da imagem; a segmentação é o processo de particionar S em r regiões R_1, R_2, \dots, R_r , tais que:

1. $\bigcup_{i=1}^r R_i = S$
2. $R_i \cap R_j = \emptyset, i \neq j$
3. R_i é uma região conexa, $\forall i = 1, \dots, r$
4. $\rho(R_i) = \text{VERDADEIRO}, \forall i = 1, \dots, r$
5. $\rho(R_i \cup R_j) = \text{FALSO}, i \neq j,$

onde $\rho(R_i)$ é um predicado lógico sobre os pixels do conjunto S , e \emptyset denota o conjunto vazio.

A primeira condição indica que a segmentação deve ser completa, ou seja, cada pixel deve pertencer a uma região. A segunda condição indica que as regiões devem ser disjuntas entre si. Estas duas primeiras condições definem o conceito de partição. A terceira condição requer que os pixels em uma região sejam conexos, como já foi explicado anteriormente. A quarta condição trata das propriedades que devem ser satisfeitas pelos pixels de uma região segmentada, por exemplo, o predicado de uma região será verdadeiro se todos os pixels dessa região possuírem a mesma intensidade. A quinta condição indica que as regiões R_i e R_j são diferentes no sentido do predicado, ou seja, elas não podem se unir por não satisfazer a condição.

Como existe uma variedade enorme de imagens, cada uma com suas propriedades e particularidades, atender a todas elas, visto os critérios estabelecidos na segmentação, não é um processo trivial, tanto que ainda não existe uma técnica considerada eficiente que segmente de forma robusta inúmeros tipos de imagens. A próxima seção expõe alguns dos problemas relacionados à segmentação de imagens.

4.1.2 Problemas

A segmentação de imagens é um dos passos mais importantes da análise automática de imagens. É nesta etapa que os objetos de interesse são extraídos da imagem para serem subsequentemente submetidos a diversas técnicas de processamento, tais como descrição e reconhecimento. Caso estes objetos de interesse não sejam extraídos na forma ideal à aplicação proposta, eles podem comprometer todo o pós-processamento.

Normalmente, o desempenho das técnicas de segmentação desenvolvidas é comparado ao sistema visual humano. Os seres humanos são um referencial de desempenho alto por seu sistema visual trabalhar de forma paralela: considerando a análise da imagem na sua totalidade, explorando a informação de profundidade, utilizando as experiências anteriores para reconhecer objetos familiares e outros. Mas, como comentado no Capítulo 1, seu potencial está restrito ao tratamento de sinais de dimensão três. Quando as imagens possuem muitas informações, como as que possuem dimensões superiores a três, pode haver falhas na segmentação manual. Esses fatos resultam num desafio à segmentação automática, pois ao mesmo tempo em que a segmentação manual pode fornecer a qualidade considerada ideal na extração dos objetos, ela é restrita.

O problema central para segmentar imagens é dividir a imagem em segmentos com atributos uniformes, interpretados como um conjunto de pixels que satisfazem a certas condições de adjacência e similaridade. Como a segmentação automática realiza este processo de forma independente, ou seja, sem um conhecimento prévio dos objetos contidos na imagem como no sistema visual humano, este problema não é trivial. Outro fator que torna o problema complexo é a ambigüidade causada, por exemplo, quando as características presentes na imagem possuem pouca diferença, ou quando a imagem apresenta ruídos.

Como existe uma diversidade de imagens e aplicações, não há um método considerado universalmente bom para segmentar imagens. A escolha de uma dada técnica depende da natureza da imagem (homogeneidade da iluminação, presença ou ausência de ruídos, importância da textura, definição de contornos e presença de oclusões dentre outras), das operações de pós-segmentação (reconhecimento de formas, interpretação, localização, medições), das primitivas a serem extraídas (contornos, segmentos retos, regiões, formas, texturas) e das limitações físicas (complexidade algorítmica, execução em tempo real, memória disponível). Os problemas associados a estas técnicas podem ser caracterizados através de duas formas principais: quando as técnicas baseam-se no domínio espacial ou das características.

Os métodos especificados no domínio espacial definem objetos diferentes baseados nas suas distâncias espaciais, especificando que pontos pertencentes a um mesmo objeto são próximos segundo algum critério de vizinhança. Este fato resulta em

segmentações com objetos espacialmente bem conectados mas sem uma garantia de que eles sejam homogêneos com relação às suas características. Os problemas associados a estes métodos são o de satisfazer as condições de similaridade com eficiência diante do problema central.

Os métodos especificados no domínio das características, como as técnicas de agrupamento de dados, assumem que os diferentes objetos em uma imagem podem ser definidos através de uma seleção realizada nas características presentes nos seus pixels, podendo estas serem o valor do nível de cinza, os valores no espaço de cores, a intensidade e outros. Para imagens que possuem objetos com características bem definidas, como ilustra a Figura 16(a), as técnicas especificadas nesta categoria possuem resultados satisfatórios, visto que, a análise nas características fornece uma visão mais natural possível à percepção humana. Se as duas bolas da Figura 16(a) possuísem características bem semelhantes, ou seja, caso a diferença entre os níveis de cinza das bolas fosse mínima como na Figura 16(b), o critério de continuidade estabelecido na segmentação poderia ser violado. Pois, mesmo que os objetos sejam separados espacialmente, eles poderiam ser considerados homogêneos perante suas características, pertencendo a um só segmento. Os problemas associados a estes métodos são em satisfazer as condições de adjacência diante do problema central.

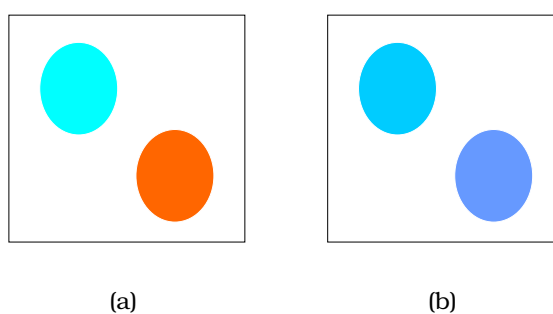


Figura 16: Diferenciação nas características da imagens: (a) bem definidas (diferentes) e (b) semelhantes.

Para consolidar a diferença entre os conceitos de domínio espacial e das características, temos a Figura 17 (página 71). Segundo o domínio espacial e estabelecendo um critério de vizinhança para este exemplo: as letras “a” e “b” pertencem a um só segmento por serem consideradas próximas espacialmente. As diferenças espaciais entre as letras “a” e “c” ou “b” e “c” são grandes diante do critério de vizinhança estabelecido. Segundo o domínio das características e estabelecendo um critério de similaridade para este exemplo, cuja característica é a forma: as letras “a” e “c” pertencem a um só segmento por possuírem semelhança na forma. As diferenças na forma entre as letras “a” e “b” ou “c” e “b” são consideradas grandes para o critério de similaridade.

A seção a seguir apresenta as modificações necessárias ao algoritmo superparamagnético, adaptando-o à segmentação de imagens vistos os problemas definidos

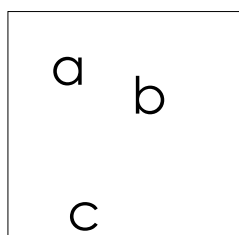


Figura 17: Domínio espacial \times domínio das características.

nesta seção.

4.2 Aplicação do algoritmo superparamagnético à segmentação de imagens

Com os conceitos definidos no Capítulo 2, no Capítulo 3 e nas seções anteriores, pode-se explicar como utilizar o método de agrupamento de dados superparamagnético para segmentar imagens. Como visto, o algoritmo superparamagnético faz uma analogia entre agrupamento de dados e ocorrências do modelo de Potts não-homogêneo. Para isto, a técnica possui três etapas, onde a primeira delas realiza a analogia física com o problema de spins de Potts. É nesta primeira etapa que alguns conceitos de segmentação são aplicados.

Para fazer a analogia física com o modelo de Potts não-homogêneo são necessários três passos: associar o conjunto de dados a ser agrupado com a técnica proposta; identificar os vizinhos de cada indivíduo de acordo com um critério pré-selecionado e calcular as interações $J_{i,j}$ entre um par de vizinhos. O primeiro passo é explicado na Seção 3.6. Já os segundo e terceiro passos são importantes para a segmentação.

Diante da discussão levantada na Seção 4.1.2, as técnicas de agrupamento de dados tendem a formar grupos através do domínio das características, sem levar em consideração o aspecto espacial. Esta propriedade do algoritmo afeta um dos critérios básicos da segmentação: a continuidade. Para que este critério seja preservado, as técnicas de agrupamento devem sofrer adaptações para que a vizinhança entre pixels também seja considerada.

Levando este novo aspecto para o algoritmo superparamagnético, tem-se que o critério pré-selecionado para identificar os vizinhos de cada indivíduo será adaptado ao conceito de vizinhança numa imagem (ver Banon 2000). Na forma original, como descreve a Seção 3.5.1, a definição de “vizinhos” é feita com relação às medidas de proximidade, onde um indivíduo é considerado vizinho ao outro se seus atributos (características) forem similares. Esta definição foi estabelecida para tornar o algoritmo flexível a qualquer tipo de dados. Porém, como no caso da segmentação os dados são dispostos na forma de uma grade regular, este algoritmo pode ser simplificado ao critério espacial para a vizinhança. Na segmentação, dois indivíduos serão

considerados vizinhos se atenderem ao seguinte critério:

$$\delta(i, j) \leq \text{Limiar},$$

onde $\delta(i, j)$ é alguma distância entre as coordenadas i e j .

Um exemplo para $\delta(i, j)$ é a distância Euclidiana. Quando o limiar para esta distância é 1, tem-se a vizinhança-4 (cada posição possui, no máximo, quatro vizinhos). Quando o limiar para esta distância vale $\sqrt{2}$, tem-se a vizinhança-8 (cada posição possui, no máximo, oito vizinhos). Os critérios de vizinhança considerados nesta dissertação foram os de vizinhança-4 e vizinhança-8 (ver Seção 2.1.4).

Definido o critério de vizinhança para o algoritmo superparamagnético aplicado à segmentação de imagens, pode-se calcular as interações locais $J_{i,j}$. Como explicado na Seção 3.5.1, as interações $J_{i,j}$ são calculadas se um indivíduo ω_i for vizinho a outro ω_j . Assim, a Equação (3.7) pode ser reescrita da seguinte forma:

$$J_{i,j} = \begin{cases} \frac{1}{K} \exp\left(\frac{-d_{ij}^2}{2a^2}\right) & \text{se } \delta(i, j) \leq \text{Limiar} \\ 0 & \text{caso contrário.} \end{cases} \quad (4.1)$$

Um comentário interessante a futuras implementações do algoritmo se faz neste parágrafo. Se as interações locais $J_{i,j}$ fossem armazenadas em uma matriz, ela possuiria a mesma estrutura que a matriz de dissimilaridade, cujo tamanho seria $n \times n$, considerando n a quantidade de pixels de uma imagem com tamanho $c \times \ell$, ou seja, $n = c\ell$. Porém, os valores atribuídos a $J_{i,j}$ nesta matriz de interações seriam influenciados diretamente pela distância $\delta(i, j)$ e pelo limiar. Assim, quanto mais restrito fosse o critério de vizinhança, mais esparsa seria a matriz.

4.3 Trabalhos Relacionados

Muitas técnicas de agrupamento de dados são utilizadas para segmentar imagens (ver, por exemplo, Sharma 2001, Lucchese & Mitra 2001). Dentre estas, a técnica de k -médias (ver Capítulo 2) é uma das mais utilizadas, apresentando inúmeras variações na sua implementação, como ocorre nas versões que a utilizam em conjunto com a métodos *fuzzy* (Harada, Miyakoshi & Shimbo 1994). Kurita (1991) apresenta uma técnica de agrupamento hierárquico para segmentar imagens por crescimento de regiões. Este algoritmo realiza o agrupamento aglomerativo tradicional, como explicado no Capítulo 2.

O algoritmo apresentado em Gdalyahu, Weinshall & Werman (2001) forma os agrupamentos hierárquicos através de processos estocásticos. A técnica baseou-se no algoritmo superparamagnético e, por isto, está bastante relacionada com o mesmo. A técnica associa o agrupamento de dados ao problema de particionar grafos, tentando resolver esta questão utilizando um algoritmo estocástico de corte mínimo.

O ECU (*Energy sharing Cluster Update*) é um algoritmo rápido e robusto para segmentar imagens utilizando o método de agrupamento superparamagnético (ver Ferber & Wörgötter 2000). Visto que o agrupamento superparamagnético original apresenta um custo computacional alto, duas otimizações são realizadas em dois passos: a formação dos grupos e a atualização dos seus spins. Uma comparação entre softwares disponíveis no mercado, sendo um deles o ECU, na segmentação de imagens de sensoriamento remoto está descrita em Neubert & Meinel (2003). Outras análises na segmentação de imagens de sensoriamento remoto podem ser encontradas em Frery, Lucca & Freitas (1999).

Em Peng et al. (2003), a segmentação de imagens pelo algoritmo superparamagnético é realizada de forma paralela para reconhecer neurônios. Já em Neirotti et al. (2003), o algoritmo superparamagnético é comparado com sistemas neurais sendo utilizada uma imagem de tomografia de um cérebro humano. Em Blatt et al. (1997), uma análise do algoritmo superparamagnético a imagens de sensoriamento remoto é apresentada.

O próximo capítulo explica o programa SPC. Além disso, para aplicar o SPC à segmentação de imagens foram necessárias tanto adaptações e melhorias no programa quanto uma interface para gerar e visualizar as segmentações realizadas. Estas melhorias e acréscimos também são detalhadas no próximo capítulo.

5 SPC - Programa e Interface

O programa SPC é a base dos trabalhos realizados nesta dissertação. Este capítulo descreve todos os procedimentos realizado por este programa. Inicialmente, o programa original é explicado através de seus arquivos de entrada e saída. Em seguida, são descritas as adaptações e melhorias realizadas visando a aplicação proposta e diante das análises feitas no programa original. Por fim, a interface desenvolvida na dissertação é apresentada.

5.1 O Programa SPC

O programa SPC (*Superparamagnetic Clustering*) (ver Blatt et al. 1996, Blatt et al. 1997, Wiseman et al. 1998, Domany et al. 1999) utilizado nesta dissertação é uma versão acadêmica que executa o algoritmo de agrupamento baseado no comportamento superparamagnético do modelo de Potts não homogêneo. Seu autor, o professor Eytan Domany, procurou desenvolver uma aplicação que satisfizesse suas necessidades imediatas, ou seja, ele precisava apenas de algum método automatizado para executar o algoritmo de agrupamento superparamagnético em grandes conjuntos de dados, e esta era uma tarefa impossível de ser realizada manualmente, devido ao tamanho dos conjuntos de dados e à complexidade dos algoritmos utilizados. Como os interesses do autor eram meramente na análise dos resultados obtidos, o SPC apresenta uma manipulação cuidadosa e bastante trabalhada, visto que é composto de arquivos de entrada e saída complicados de serem gerados e analisados a medida em que o conjunto de dados aumenta. Em casos de grandes conjuntos de dados, como as imagens, a criação dos dados de entrada só pode ser realizada de forma automatizada. Da mesma forma, as possíveis análises dos resultados só podem ser realizadas após um pós-processamento dos arquivos de saída.

A versão acadêmica do SPC foi desenvolvida em linguagem C e só pode ser executada por linha de comando. Este programa necessita de dois arquivos de entrada para ser executado, e origina cinco arquivos de saída contendo várias informações sobre o agrupamento superparamagnético como, por exemplo, os próprios grupos formados (ver Figura 18, página 75).

As próximas subseções explicam os arquivos de entrada e saída utilizados pelo

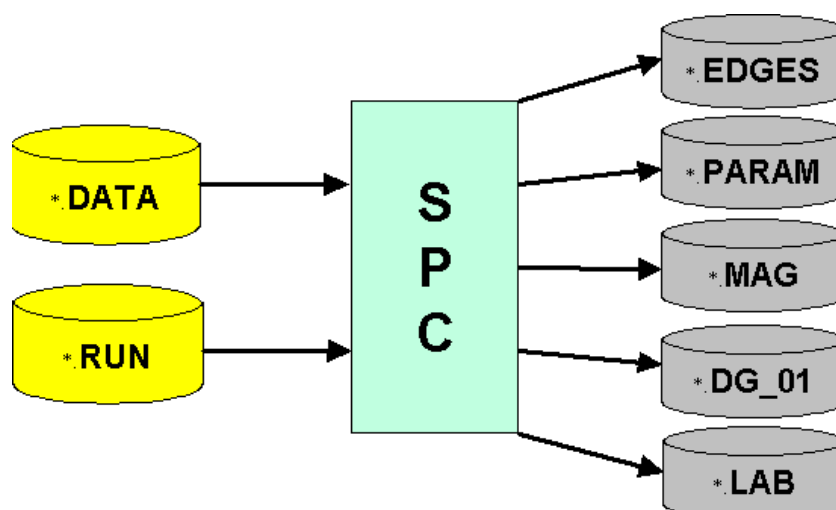


Figura 18: Estrutura original do SPC.EXE.

SPC. Serão explicadas também as modificações realizadas para uma otimização no desempenho de todo o processo, desde a geração dos arquivos à análise dos resultados.

5.1.1 Estrutura dos Arquivos de Entrada

Como ilustra a Figura 18, o SPC necessita de dois arquivos de entrada: o arquivo de informações sobre os dados de entrada, cuja extensão é `.DATA`, e o arquivo de parâmetros. Este último fornece tanto algumas informações sobre o arquivo de dados, quanto parâmetros necessários à execução do programa, e sua extensão é `.RUN`.

O arquivo de dados pode apresentar duas formas: os valores originais dos dados ou a matriz de dissimilaridade gerada a partir deste conjunto de dados. Na primeira opção cada indivíduo está representado por uma linha do arquivo, sendo as variáveis dispostas em colunas separadas por espaços para cada indivíduo. Um exemplo disto está ilustrado na Tabela 14 (página 76), considerando os dados da Figura 7 (página 45). Quando este tipo de arquivo é utilizado, deve-se atribuir ao parâmetro *Dimentions* do arquivo com extensão `.RUN` a quantidade de variáveis disponíveis no arquivo de dados. No exemplo, *Dimentions* é 3. Como a matriz de dissimilaridade proveniente dos dados de entrada é utilizada no programa, quando a entrada são os dados reais há a necessidade de sua construção pelo programa. A desvantagem decorrente do uso deste tipo de arquivos de entrada é a limitação associada à medida de dissimilaridade, que é fixada pelo programa. Quando a própria matriz de dissimilaridade é fornecida no arquivo de entrada esta opção torna-se flexível, visto as inúmeras medidas de dissimilaridade disponíveis, como algumas apresentadas na Seção 2.3. Visando esta flexibilidade e otimização no programa, a dissertação se concentrou em trabalhar somente com arquivos de dados que fornecem como entrada a matriz de dissimilaridade.

255	255	255
0	0	255
0	255	0
255	255	255
0	0	0
0	255	0
255	0	0
0	0	255
0	255	0

Tabela 14: Arquivo de dados com os dados reais.

Os arquivos de dados também podem conter a matriz de dissimilaridade dos dados a serem agrupados. Ela pode ser calculada por algum software comercial como, por exemplo a plataforma R. Quando a matriz de dissimilaridade é fornecida pelo arquivo de dados, o parâmetro *Dimentions* do arquivo com extensão *.RUN* é 0. Quando este parâmetro tem o valor 0, o programa não constrói a matriz de dissimilaridade, apenas a utiliza pela captura do arquivo de entrada. A construção de uma matriz de dissimilaridade é explicada na Seção 2.3. Um exemplo deste tipo de arquivo está ilustrado na Tabela 15, considerando os dados da Figura 6 (página 44) e a matriz de dissimilaridade da Tabela 11 (página 48). A figura da imagem monocromática será considerada neste exemplo por sua matriz de dissimilaridade possuir um tamanho pequeno.

0
0
255
255
0
0
255
255
255
255
0
0
255
255
0
0

Tabela 15: Arquivo de dados com a matriz de dissimilaridade.

O segundo arquivo de entrada para o SPC possui a extensão *.RUN* e contém informações necessárias à execução do programa, além de informações sobre a matriz de dissimilaridade do arquivo *.DATA*. Um exemplo do arquivo *.RUN* é ilustrado na Tabela 16 (página 77), baseando-se nos dados da Figura 7 (página 45).

NumberOfPoints:	9
DataFile:	C:\Image.data
Dimensions:	0
MinTemp:	0.00
MaxTemp:	0.15
TempStep:	0.005
OutFile:	ImgResult
SWCycles:	4000
KNearestNeighbours:	5
MSTree	
DirectedGrowth	
SaveSuscept	
WriteLabels	
DataIsMatrix	
WriteCorFile	~

Tabela 16: Campos do arquivo com extensão .RUN.

Alguns dos campos da Tabela 16 podem ser facilmente identificados, tais como *NumberOfPoints*, que representa o número de indivíduos presentes no conjunto de dados, neste exemplo são 9 pixels. *DataFile* contém o caminho e o nome do arquivo contendo a matriz de dissimilaridade. *Dimensions* indica a dimensão dos vetores de variáveis (atributos) que descrevem os indivíduos, se seu valor é 0 (zero) então uma matriz de dissimilaridade é esperada. *MinTemp* contém a temperatura inicial a partir da qual os agrupamentos serão realizados. *MaxTemp* contém a temperatura final até onde os agrupamentos serão realizados. *TempStep* representa o incremento da temperatura, ou seja, a partir da temperatura inicial o valor de *TempStep* será adicionado ao valor atual da temperatura até que a temperatura final seja atingida. *OutFile* é um prefixo para os arquivos de saída. *SWCycles* é o número de ciclos a ser considerado no algoritmo de *Swendsen-Wang*. *KnearestNeighbours* estipula o número máximo de vizinhos considerados a cada indivíduo. Como explicado no Capítulo 3, o algoritmo superparamagnético necessita identificar vizinhos a cada indivíduo. No programa SPC, a indentificação destes vizinhos é realizada pelo algoritmo KNN. O KNN (*K-Nearest-Neighbour*) é uma técnica que identifica os *K* vizinhos considerados mais próximos de um indivíduo de acordo com algum critério de proximidade. Neste caso, o critério de proximidade está definido pela matriz de dissimilaridade apresentada. Com o parâmetro *KnearestNeighbours*, o usuário pode determinar a quantidade máxima de vizinhos. Assim como os parâmetros explicados acima, *MSTree*, *DirectedGrowth*, *SaveSuscept*, *WriteLabels*, *DataIsMatrix* e *WriteCorFile* são funções que geram os arquivos de saída do SPC e que manipulam os dados obtidos do algoritmo superparamagnético.

5.1.2 Estrutura dos Arquivos de Saída

Como ilustrados na Figura 18 (página 75), os arquivos de saída do SPC são cinco, mas, neste trabalho, foram considerados apenas dois, por possuírem as informações

necessárias na análise dos dados agrupados. Esses dois arquivos são .DG_01 e .LAB, que armazenam os dados resultantes dos agrupamentos, tais como os grupos e os indivíduos pertencentes a cada grupo.

Os outros três arquivos de saída armazenam informações de controle, onde o usuário pode verificar a corretude das operações realizadas pelo programa. O arquivo com extensão .EDGES é um arquivo opcional e é usado para forçar o programa SPC a considerar bordas específicas, além das selecionadas pelo método KNN. Cada borda é descrita por um alinhamento no formato:

```
i1 i2,
```

onde i1 e i2 são os índices dos pontos.

O arquivo com extensão .PARAM contém uma lista dos parâmetros usados pelo programa. Alguns desses parâmetros foram explicados na seção anterior. A Tabela 17 exibe um exemplo desse tipo de arquivo. O arquivo com extensão .MAG é um arquivo de *log* que armazena os passos de execução do SPC.

AverageInteraction	0.074233
ClustersReported	12
CharDist	13.567979
DataFile	yoram.data
Dimentions	0
DirectedGrowth	
DataIsMatrix	
Dimensions	0
KNearestNeighbours	10
MinTemp	0.00
MaxTemp	0.15
MSTree	
NumberOfPoints	90
NumberOfEdges	375
NearestNeighbrs	8.333333
OutFile	yout
PottsSpins	20
RandomSeed	951312526
SusceptColors	4
SaveSuscept	
SWCycles	4000
SWFraction	0.800000
ThresholdTheta	0.500000
TempStep	0.005
WriteLabels	

Tabela 17: Exemplo de arquivo com extensão .PARAM.

Como os autores projetaram o SPC para grandes quantidades de dados, determinou-se uma quantidade mínima de *NumberOfPoints*, no caso 12. Como os exemplos

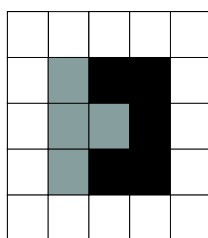


Figura 19: Imagem monocromática 5×5 .

considerados até agora na ilustração dos arquivos de entrada só possuem *NumberOfPoints* inferiores a 12, será definido um novo exemplo para ser utilizado com o programa. Considere a imagem monocromática da Figura 19. Ela é composta de três níveis de cinza: 255 indicando o branco, 0 indicando o preto e 128 indicando o cinza. Foi criada uma matriz de dissimilaridade através da distância Euclidiana para gerar o arquivo de dados. O arquivo com extensão `.RUN` está ilustrado na Tabela 18.

NumberOfPoints:	25
DataFile:	C\Image5.data
Dimensions:	0
MinTemp:	0.000
MaxTemp:	0.020
TempStep:	0.005
OutFile:	ImgResult5
SWCycles:	4000
KNearestNeighbours:	12
MSTree	
DirectedGrowth	
SaveSuscept	
WriteLabels	
DataIsMatrix	
WriteCorFile	~

Tabela 18: Arquivo com extensão `.RUN` da imagem monocromática 5×5 .

O arquivo com extensão `.DG_01` apresenta a estrutura ilustrada na Tabela 19 (página 80). O primeiro campo da tabela indica os índices das temperaturas consideradas e está variando de 0 - 4. O segundo campo indica o valor da temperatura utilizada, variando, neste caso, de 0.00000 - 0.02000. O terceiro campo, que contém apenas valores 0 (zero), indica o número mínimo de indivíduos que estão fazendo parte de algum cluster àquela temperatura, nesse caso existem clusters que não contêm indivíduos. O quarto campo, que possui os valores {1,3}, representa o total de clusters que possuem pelo menos um indivíduo. Os outros campos da tabela, 12 no total, são os clusters obtidos em cada temperatura. Foi estipulado pelos desenvolvedores do SPC a exibição de no mínimo 12 cluster, mesmo que estes não existam, casos em que o número de indivíduos é 0. Analisando a Tabela 19 (página 80), na temperatura 0.00000 só há um cluster que é o considerado pelo índice 0 cuja quantidade de indivíduos é 25. Na temperatura 0.00500 encontram-se três grupos: o cluster 0 com 16

indivíduos, o cluster 1 com 5 indivíduos e o cluster 2 com 4 indivíduos. O resultado obtido na temperatura 0.00500 repete-se nas temperaturas 0.01000, 0.01500 e 0.02000, indicando que os grupos permaneceram com a mesma estrutura durante estas quatro temperaturas, ou seja, não aconteceram mais agrupamentos após a temperatura 0.00500.

0	0.00000	0	1	25	0	0	0	0	0	0	0	0	0	0
1	0.00500	0	3	16	5	4	0	0	0	0	0	0	0	0
2	0.01000	0	3	16	5	4	0	0	0	0	0	0	0	0
3	0.01500	0	3	16	5	4	0	0	0	0	0	0	0	0
4	0.02000	0	3	16	5	4	0	0	0	0	0	0	0	0

Tabela 19: Arquivo com extensão DG_01.

O arquivo com extensão .LAB também armazena os agrupamentos como o arquivo com extensão .DG_01, mas de um ponto de vista diferente. Ao invés de atribuir a cada cluster a quantidade de indivíduos presentes, o arquivo associa a cada indivíduo o índice do cluster respectivo. A Tabela 20 ilustra este ponto de vista.

0	0.00000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0.00500	0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0	0
2	0.01000	0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0	0
3	0.01500	0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0	0
4	0.02000	0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0	0

Tabela 20: Arquivo com extensão .LAB.

A Tabela 20 representa o arquivo com extensão .LAB correspondente ao arquivo com extensão .DG_01. Pode-se verificar que os dois primeiros campos desse arquivo são idênticos aos do com extensão .DG_01, enquanto os outros, no total de 25, representam agora os indivíduos e não mais os clusters. Analisando os outros 25 campos da Tabela 20, temos que na temperatura 0.00000 todos os indivíduos pertencem ao cluster 0 (zero), por isso a representação 0 nos campos, ou seja, as posições representam os indivíduos e os valores dos campos os clusters aos quais os indivíduos pertencem. Na temperatura 0.00500, os indivíduos nas posições 6, 11, 12 e 16 pertencem ao cluster 0, considerando o primeiro indivíduo na posição 0. Os indivíduos nas posições 7, 8, 13, 17 e 18 pertencem ao cluster 1 e os demais indivíduos pertencem ao cluster 0. Calculando-se o total de indivíduos por cluster no arquivo com extensão .LAB, obtém-se exatamente o arquivo com extensão .DG_01. Considerando os resultados apresentados na temperatura 0.00500 da Tabela 20 e mapeando os índices correspondentes a esta temperatura com os pixels de uma imagem matricial, tem-se definição dos três grupos apresentados na Figura 20 e provenientes da imagem original da Figura 19 (página 79).

Diante dos arquivos de entrada e saída do SPC mostrados nessas duas seções, percebe-se o quão complicado é gerar e manusear os mesmos para grandes volumes

0	0	0	0	0
0	2	1	1	0
0	2	2	1	0
0	2	1	1	0
0	0	0	0	0

Figura 20: Imagem resultante definida por números.

de dados, como é o caso de imagens. Além disso, a ativação do sistema é realizada por linha de comando. Estes fatos resultam em muitas dificuldades para qualquer usuário, mesmo aos que entendam de computação. Visando a interação usuário-programa e a aplicação proposta, foi desenvolvida uma interface para trabalhar em conjunto com o programa *SPC*. Em adição, foram realizadas algumas adaptações tanto nos arquivos de entrada, quanto nos de saída e, também, no próprio programa para otimizar alguns dos seus procedimentos e aplicá-lo à segmentação de imagens. Estas alterações e a interface são descritas nas seções seguintes.

5.2 Otimizações

Com o objetivo de adaptar o programa *SPC* à segmentação de imagens e ao desenvolvimento da interface, inicialmente, realizou-se uma análise em todo o desempenho do programa. Observou-se a necessidade de algumas modificações e melhorias em algumas partes deste processo. Estas otimizações são explicadas nesta seção e serão subdivididas em três etapas: uma para os arquivos de entrada, outra para os arquivos de saída e outra para a adaptação à segmentação de imagens. As subseções seguintes apresentam estas alterações.

5.2.1 Arquivos de Entrada

Como descrito na Seção 5.1.1, o arquivo de dados considerado nesta dissertação informa a matriz de dissimilaridade do conjunto de dados. Para um conjunto de dados com poucas informações, a construção do arquivo é rápida. Quando o conjunto possui muitos dados, como as imagens, esta construção é custosa computacionalmente, com o seu custo proporcional ao tamanho da imagem.

Para explicar este fato, considere que o conjunto de dados é uma imagem de tamanho 300×400 . Segundo as explicações feitas no Capítulo 2, o número de indivíduos (pixels) desse conjunto é $n = 12000$, visto que $n = cl$. Como a matriz de dissimilaridade é $n \times n$, ela terá um tamanho de 12000×12000 , ou seja, para cada um dos 12000 pixels serão calculadas 12.000 medidas de dissimilaridade. Neste exemplo, o arquivo de dados original conterà $144 \cdot 10^6$ informações para serem acessadas pelo programa.

Considerando algumas propriedades contidas na matriz de dissimilaridade e estabelecidas pela aplicação proposta, pode-se reduzir significativamente a quantidade de processos executados para geração do arquivo de dados.

Inicialmente, será analisada uma propriedade de matrizes de dissimilaridade baseadas em distâncias: a simetria. O valor da dissimilaridade calculado na posição (i, j) é o mesmo para o da posição (j, i) , ou seja, $d_{ij} = d_{ji}$. Um exemplo está ilustrado na matriz de dissimilaridade da Tabela 13 (página 49), percebe-se que $d_{12} = d_{21} = 361$. Esta propriedade fornece uma matriz com informações redundantes, podendo as dissimilaridades calculadas entre o conjunto de dados serem dispostas apenas por uma matriz triangular. No caso da matriz ilustrada na Tabela 13, pode-se representá-la pela matriz da Tabela 21. Construindo o arquivo de dados com a matriz triangular ao invés da matriz completa, há uma redução significativa, a quase a metade do tamanho original, da quantidade de informações a serem calculadas e acessadas pelo programa, tendo, também, como consequência uma redução no tamanho do arquivo. No exemplo, a matriz completa possui 81 informações para serem armazenadas no arquivo de dados, já a matriz triangular possui 45 informações.

k/k	1	2	3	4	5	6	7	8	9
1	0	361	361	0	442	361	361	361	361
2		0	361	361	255	361	361	0	361
3			0	361	255	0	361	361	0
4				0	442	361	361	361	361
5					0	255	255	255	255
6						0	361	361	0
7							0	361	361
8								0	361
9									0

Tabela 21: Matriz de dissimilaridade triangular.

Analisando uma das propriedades da segmentação de imagens, como descreve a Seção 4.1.1 (página 67), tem-se que em uma região cada pixel deve estar conectado a todos os outros pertencentes à região, mesmo que estes estejam conectados através de um caminho. Para que esta propriedade prevaleça, a análise da dissimilaridade deve ser mais evidente nos vizinhos de um determinado pixel do que em outros considerados distantes por um critério de vizinhança, ou seja, deve-se calcular a dissimilaridade apenas entre um pixel e seus vizinhos. Um exemplo interessante está ilustrado na Figura 7 (página 45). Se as dissimilaridades fossem calculadas entre o pixel na posição $(1, 0)$ (ordenada por $(coluna, linha)$) e todos os outros pixels, para a imagem resultante esperada, seria muito provável o agrupamento entre este pixel (cor azul) com o pixel de posição $(1, 2)$, também de cor azul porém distante espacialmente, comprometendo a “continuidade” da segmentação resultante. Se o critério de vizinhança-4 fosse aplicado ao cálculo da dissimilaridade, estes pixels não seriam agrupados na imagem resultante esperada.

Diante desta análise, o processo de construção da matriz de dissimilaridade é reduzido substancialmente. Ao invés das dissimilaridades serem calculadas entre cada um dos n pixels com todos os outros n pixels, elas são calculadas entre cada um dos n pixels com os seus vizinhos de acordo com algum critério de vizinhança, tendo a dissimilaridade um valor infinito para os pixels que não são vizinhos ao pixel analisado ou, em outras palavras, pixels que não são espacialmente contíguos terão uma similaridade nula independentemente do valor observado neles.

Neste trabalho, a vizinhança-4 e a vizinhança-8 foram consideradas. Aplicando o critério de vizinhança-4 para a matriz de dissimilaridade completa da Tabela 13 (página 49), tem-se a matriz da Tabela 22. Neste exemplo, o processo de calcular a dissimilaridade é reduzido de 81 para 33 cálculos.

k/k	1	2	3	4	5	6	7	8	9
1	0	361	∞	0	∞	∞	∞	∞	∞
2	361	0	361	∞	255	∞	∞	∞	∞
3	∞	361	0	∞	∞	0	∞	∞	∞
4	0	∞	∞	0	442	∞	361	∞	∞
5	∞	255	∞	442	0	255	∞	255	∞
6	∞	∞	0	∞	255	0	∞	∞	0
7	∞	∞	∞	361	∞	∞	0	361	∞
8	∞	∞	∞	∞	255	∞	361	0	361
9	∞	∞	∞	∞	∞	0	∞	361	0

Tabela 22: Matriz de dissimilaridade com critério de vizinhança-4.

Estes dois processos de otimização da matriz de dissimilaridade são utilizados nesta dissertação. A junção é realizada aplicando o critério de vizinhança estabelecido à matriz triangular a ser construída. Com a aplicação do critério de vizinhança-4 para a matriz de dissimilaridade da Tabela 21 (página 82), tem-se a matriz da Tabela 23. Neste exemplo, o processo de calcular a dissimilaridade é reduzido de 45 para 21 cálculos.

Fazendo a comparação com a matriz completa, tem-se que ao invés de se armazenar 81 informações no arquivo de dados, só são armazenadas 21 informações.

k/k	1	2	3	4	5	6	7	8	9
1	0	361	∞	0	∞	∞	∞	∞	∞
2		0	361	∞	255	∞	∞	∞	∞
3			0	∞	∞	0	∞	∞	∞
4				0	442	∞	361	∞	∞
5					0	255	∞	255	∞
6						0	∞	∞	0
7							0	361	∞
8								0	361
9									0

Tabela 23: Matriz de dissimilaridade triangular com critério de vizinhança-4.

O formato do arquivo de dados otimizado é bem diferente do arquivo original. Como explicado no parágrafo anterior, o arquivo otimizado contém a matriz de dissimilaridade triangular, cujos cálculos de dissimilaridade só ocorrem entre pixels vizinhos, como é o exemplo da matriz ilustrada na Tabela 23 (página 83). Já o arquivo original contém a matriz completa.

No arquivo original, como todas as dissimilaridades são fornecidas, o acesso a cada posição desejada pode ser feita de forma direta. No arquivo otimizado, o acesso à posição desejada torna-se mais complicado. Por convenção e facilidade no acesso, os dois pixels correspondentes ao cálculo de cada dissimilaridade são fornecidos no arquivo otimizado. O formato do arquivo de dados otimizado é ilustrado na Tabela 24. Para o exemplo da Tabela 23, o formato do arquivo de dados otimizado está ilustrado na Tabela 25 (página 85).

```

número de atributos da imagem
índices dos atributos (utilizados no cálculo da dissimilaridade)
quantidade total de indivíduos ( $n$ )
largura ( $c$ )
altura ( $l$ )
pixel  $a$ 
pixel  $b$ 
 $d_{ab}$ 
pixel  $c$ 
pixel  $e$ 
 $d_{ce}$ 
...
pixel  $x$ 
pixel  $z$ 
 $d_{xz}$ 

```

Tabela 24: Arquivo de dados otimizado.

É importante diferenciar o acesso aos dados de entrada e a manipulação dos mesmos no programa. O acesso aos dados de entrada só acontece uma única vez, tornando-os disponíveis ao programa. A otimização foi realizada para acelerar este processo tanto na sua geração quanto na leitura do arquivo de entrada pelo programa. A forma como o programa armazena e manipula estes dados não foi otimizada neste trabalho, visto que os objetivos principais deste trabalho são aplicar a segmentação de imagens e analisar os dados de saída do SPC. Como a otimização na manipulação dos dados não foi realizada, o programa armazena os dados utilizando a matriz de dissimilaridade completa. Este fato limita a utilização do SPC a imagens com tamanhos pequenos dependendo das configurações apresentadas no computador a ser trabalhado. Uma sugestão para trabalhos futuros é otimizar esta manipulação no SPC resolvendo o problema de matrizes esparsas. Outra possibilidade é trabalhar com uma estrutura que armazene a matriz de forma triangular, visto que a matriz triangular é fornecida pelo arquivo otimizado com extensão .DATA.

```

3
0 1 2
9
3
3
1
1
0
1
2
361
1
4
0
...
8
9
361
9
9
0

```

Tabela 25: Exemplo de arquivo de dados otimizado.

5.2.2 Arquivos de Saída

Ao iniciar o desenvolvimento da interface foi verificado que os cinco arquivos de saída do programa SPC (ver Figura 18, página 75) não eram necessários para realizar a análise de agrupamento de dados, mas apenas os dois arquivos mencionados na Seção 5.1.2, ou seja, os de extensões `.DG_01` e `.LAB`. Diante das análises realizadas nestes dois arquivos, observou-se que algumas informações são redundantes e dificultavam a construção dos resultados tornados disponíveis pela interface, como é o caso dos dendrogramas. Uma otimização nestes arquivos foi necessária para deixá-los apenas com as informações relevantes e evitar que um pré-processamento fosse necessário nestes arquivos originais.

O arquivo com extensão `.LAB` sofreu duas mudanças. A primeira foi na ordem com que as temperaturas e seus respectivos agrupamentos são armazenados. No arquivo original, os agrupamentos são exibidos da temperatura inicial até a final, como pode ser visto na Tabela 20 (página 80). No arquivo otimizado, a ordem inversa foi adotada (ver Tabela 26, página 86). Esta primeira alteração foi necessária por causa do algoritmo utilizado na construção dos dendrogramas, uma vez que ele estrutura o gráfico da temperatura com maior valor para a de menor valor. Com esta mudança, ganha-se tempo de processamento, uma vez que não é mais necessário o armazenamento de todos os agrupamentos para em seguida montar o dendrograma. Dessa maneira, à medida que os agrupamentos são lidos, as estruturas dos dendrogramas são montadas. O SPC define os agrupamentos da temperatura mais baixa para a

mais alta, como a ordem inversa é necessária, estes agrupamentos são inicialmente armazenados em uma estrutura de dados dinâmica, para em seguida serem armazenados em arquivo. Uma lista duplamente encadeada foi utilizada para armazenar os agrupamentos e as informações que serão definitivamente salvas em arquivos e lidas posteriormente pela interface.

4	0.02000	0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0
3	0.01500	0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0
2	0.01000	0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0
1	0.00500	0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0
0	0.00000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabela 26: Arquivo com extensão .LAB com ordem invertida.

A segunda alteração no arquivo com extensão .LAB melhorou a performance do programa e permitiu que os arquivos gerados ocupassem menos espaço em disco. Como foi visto na Tabela 20 (página 80), em algumas temperaturas não ocorrem novos agrupamentos e assim os agrupamentos anteriores são repetidos. A segunda alteração encarrega-se de eliminar estas repetições e armazenar no arquivo apenas as linhas em cujas temperaturas aconteceram novos agrupamentos (ver Tabela 27). Como pode ser observado na Tabela 27, uma repetição ainda é mantida, isto acontece porque, por convenção, preferiu-se manter nos arquivos de saída os agrupamentos das temperaturas mais baixa e mais alta, para facilitar a representação durante o processamento da interface. Mesmo sendo mantida esta repetição, o ganho no armazenamento é considerável e a simplicidade da estrutura do arquivo de saída é mantida.

Outra modificação introduzida digna de menção foi a eliminação dos índices que representam a temperatura e seus respectivos valores onde, para cada linha, o arquivo alterado só apresenta os grupos associados a cada indivíduo.

0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	2	1	1	0	0	2	2	1	0	0	2	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabela 27: Arquivo com extensão .LAB adaptado.

O programa SPC apresenta uma limitação que também foi retirada, ela refere-se ao número de grupos armazenados nos arquivos. Este número de grupos é fixo, ou seja, é uma constante configurada no código fonte do programa. Esta restrição no número de grupos não era interessante para os objetivos da interface, cujo intuito é mostrar todos os agrupamentos realizados. Com o número reduzido de grupos isso não era possível, uma vez que a hierarquia dos agrupamentos não era mostrada. Essa mudança tornou os arquivos de saída mais completos, ou seja, eles passaram a armazenar as informações de todos os agrupamentos realizados.

O segundo arquivo, de extensão .DG_01, também teve sua ordem de armazena-

mento invertida e as informações redundantes eliminadas. Nele houve um acréscimo de informações sobre os agrupamentos e as propriedades usadas na geração do arquivo com extensão `.DATA`. Cinco novos parâmetros foram introduzidos nas primeiras linhas desse arquivo, são eles:

1. número de atributos da imagem utilizados na geração do arquivo `.DATA`;
2. índices dos atributos utilizados na geração do arquivo `.DATA`;
3. número de temperaturas onde ocorreram novos agrupamentos;
4. total de grupos gerados durante o agrupamento;
5. número de indivíduos utilizados nos agrupamentos.

Os atributos da imagem representam os componentes ou bandas que definem a imagem. Quando esta quantidade é 3, por exemplo, pode-se representar uma imagem colorida pelo sistema RGB. Como o algoritmo implementado para construção da matriz de dissimilaridade permite a escolha dos atributos que serão utilizados na construção da mesma, estes atributos podem ser identificados pela interface e informados ao usuário. Para isto, os atributos devem ser fornecidos aos arquivos de saída. O primeiro parâmetro indica a quantidade destes atributos enquanto o segundo informa os índices que representam os atributos utilizados. O terceiro parâmetro indica a quantidade de temperaturas às quais os respectivos agrupamentos exibem alguma modificação, ou seja, o número de temperaturas a serem armazenadas no arquivo. O quarto parâmetro descreve a quantidade total de grupos formados, e o último parâmetro indica a quantidade total de pixels da imagem. A Tabela 28 ilustra o arquivo com extensão `DG_01` com as adaptações necessárias à interface implementada, dado que o arquivo original está ilustrado na Tabela 19 (página 80).

3					
0	1	2			
3	3	25			
4	0.02000	16	5	4	
1	0.00500	16	5	4	
0	0.00000	25	0	0	

Tabela 28: Arquivo com extensão `DG_01` alterado.

5.2.3 Segmentação de Imagens

O programa `SPC` define os vizinhos de um indivíduo (pixel) através do algoritmo `KNN` (K-Nearest Neighbour). Este critério de vizinhança encontra os vizinhos através de medidas de proximidade. Este fato acarreta nos problemas comentados na Seção 4.1.2 (página 69): a descoberta dos vizinhos é realizada no espaço das características e não

no domínio espacial. Quando o critério envolve apenas as características, a conectividade entre os pixels de um segmento pode ser afetada. Quando o domínio espacial é estabelecido, esta propriedade é garantida, visto que os agrupamentos só ocorrem entre pixels próximos espacialmente.

Como a técnica KNN está baseada no domínio das características, deve-se atribuir o domínio espacial em alguma parte do seu processo. Isto pode ser realizado na construção da matriz de dissimilaridade (entrada do KNN no programa), visto que o critério de vizinhança espacial pode ser aplicado na mesma (ver Seção 5.2.1). Ao mesmo tempo, esta alternativa não garante que os vizinhos encontrados pelo método sejam os ideais segundo o domínio espacial. Como o arquivo de dados otimizado aplica o critério de vizinhança-4 ou vizinhança-8 à matriz de dissimilaridade construída, a técnica utilizada pelo programa pode ser aplicada à segmentação de imagens.

Outro fator que influencia a utilização desta técnica é o seu custo computacional. No KNN, para cada pixel devem ser encontrados k vizinhos mais próximos segundo alguma medida de proximidade. No programa SPC, este processo já possui uma otimização, dado que a medida de proximidade já foi calculada pela entrada do algoritmo ser uma matriz de dissimilaridade. Porém, só a descoberta dos k vizinhos torna-se um processo custoso quando o volume de dados é grande. Como as imagens são conjuntos de dados de grande volume, a utilização desta técnica torna a execução do SPC bastante lenta. Se o critério de vizinhança estabelecido fosse fixo, como é o caso de vizinhança-4 ou vizinhança-8, a descoberta dos vizinhos sofreria uma redução bastante significativa no tempo de execução.

Diante das desvantagens encontradas na utilização de algoritmo KNN para a segmentação de imagens, o programa SPC sofreu alterações no seu critério de vizinhança: foi implementada a vizinhança-8 como critério adicional para o cômputo das dissimilaridades. Como descrito anteriormente, os critérios de vizinhança considerados nesta dissertação são os de vizinhança-4 e vizinhança-8. Segundo a Seção 2.1.4 (página 29), a vizinhança-8 contém a vizinhança-4, por isto ela foi escolhida na implementação.

Após essas alterações no SPC e arquivos associados ao mesmo, o projeto e implementação da interface puderam ser iniciados. Estas fases são descritas nas próximas seções.

5.3 Interface

A versão acadêmica do programa SPC, como descrito na Seção 5.1, apesar de apresentar bons resultados e ser eficiente na realização dos agrupamentos de dados, não permite uma interação simples e amigável a qualquer usuário. Quando o conjunto de dados possui um grande volume de dados, a manipulação do mesmo se torna muito trabalhosa ou até inviável. Em particular, no caso de imagens, há a necessidade de um pós-processamento, visto que os resultados obtidos são impossíveis

de serem entendidos só pela análise dos arquivos de saída. Para facilitar toda essa manipulação, foi desenvolvida uma interface amigável que procura extrair uma variedade de informações para a análise completa dos resultados provenientes do SPC. A interface foi desenvolvida em IDL e ENVI (ver Seções 1.3.1 e 1.3.2, página 20). A escolha dessa linguagem e software foi feita visto as vantagens encontradas, onde algumas delas são: facilidade na manipulação de imagens, muitos recursos estatísticos e na manipulação de matrizes. Enquanto esta dissertação trabalhou com um sistema aplicado à segmentação de imagens, em Oliveira (2004), tem-se uma aplicação a dados com atributos multivariados como, por exemplo, da estatística, biologia e outros.

O desenvolvimento de uma interface amigável para o programa SPC é uma maneira de verificar a nova técnica de agrupamento de dados e, mais ainda, de permitir um maior aproveitamento dos resultados obtidos e da maneira mais simples possível. Buscando antigir essa simplicidade e esse maior aproveitamento, foi realizada uma análise dos requisitos do sistema, com o objetivo de definir a priori tudo o que seria necessário para o desenvolvimento de uma interface amigável, completa e simples de usar por qualquer usuário.

Para realizar essa análise foi utilizada uma abordagem mista, envolvendo ferramentas de modelagem de objetos (ver Furlan 1998) e de análise essencial (ver Pompilho 2002, Yourdan 1990), para exibir de maneira clara o significado de cada função componente do sistema.

As próximas seções exibem tanto a análise dos requisitos e suas funções, enfocando os eventos do sistema, quanto a implementação do sistema com base na seção de análise.

5.3.1 Análise e Projeto dos Requisitos

A especificação de requisitos para o desenvolvimento de uma interface é uma tarefa complicada, uma vez que interfaces são orientadas por eventos, que é um mecanismo conhecido como estímulo/resposta, ou seja, o usuário produz um estímulo dependendo de sua necessidade e dos recursos disponibilizados pela interface e esta, por sua vez, emite uma resposta ou realiza algum processamento. A parte complicada de se trabalhar com sistemas baseados em eventos é especificar o que será e o que não será útil para os usuários que utilizarão o sistema, sem confundí-los com a inserção de opções desnecessárias.

Podemos classificar os eventos em três tipos diferentes, são eles:

Evento orientado por fluxo de dados: é aquele em que o estímulo é a chegada ao sistema de um fluxo de dados enviado por uma entidade externa.

Evento orientado por controle: é aquele em que o estímulo é a chegada ao sistema

de um fluxo de controle, por exemplo a ativação de alguma variável binária.

Evento orientado por tempo: é aquele em que o estímulo é a chegada ao sistema da informação de haver passado um determinado intervalo de tempo.

A implementação da interface foi baseada na análise de seus eventos e no controle do sistema. Como dito antes, a metodologia utilizada para realizar a análise corresponde a técnicas e ferramentas de análise essencial e de modelagem orientada a objetos. A análise essencial envolve três abordagens, das quais apenas duas são usadas aqui, são elas as abordagens **funcional** e de **controle**. No início da análise do sistema aqui proposto foi utilizada uma sobreposição das técnicas de análise essencial e de modelagem de dados, definindo o sistema a partir de suas funções e representando esse comportamento através de diagramas de respostas a eventos e diagramas de caso de uso. Uma visão mais geral do sistema pôde ser obtida a partir da modelagem de objetos, exibindo as interações do usuário com os módulos/funções componentes do sistema e as interações entre elas próprias.

A Tabela 29 exhibe uma lista de eventos do sistema classificando-os da seguinte maneira:

- (F) - Evento orientado por fluxo de dados
- (T) - Evento temporal
- (C) - Evento orientado por controle

Estes eventos são considerados os requisitos principais do sistema e é a partir deles que decorre todo desenvolvimento do projeto.

Número	Nome	Tipo
1	Abrir imagem	(F)
2	Fechar imagem	(C)
3	Visualizar imagem	(C)
4	Fornecer MDA	(C)
5	Gerar MD	(C)
6	Selecionar Algoritmo Agrupamento	(F)
7	Visualizar imagem segmentada	(F)
8	Gerar dendrograma	(F)
9	Procurar grupos	(F)
10	Gerar animações	(F)

Tabela 29: Principais eventos do sistema.

Os eventos da Tabela 29 representam as principais funções do sistema, dentre estes destacam-se os eventos com numeração 4, 7, 8, 9 e 10, que são responsáveis pela integração do módulo principal do sistema com os módulos de análise multivariada dos dados, visualização da imagem segmentada, dendrogramas, procurar grupos e



Figura 21: Caso de uso geral para o Sistema Proposto.

geração de animações, repectivamente. De uma maneira geral o sistema pode ser representado como na Figura 21.

De acordo com o diagrama de caso de uso apresentado na Figura 21, pode-se verificar que o usuário e o programa SPC são as únicas entidades externas (atores) que se comunicam com a interface do sistema. Investigando esta interface, podemos representar, de uma forma mais detalhada, as funções do sistema de acordo com a Figura 22 (página 92).

As funções apresentadas na Figura 22 são descritas nas próximas seções, bem como as ações geradas para acioná-las e suas respostas aos estímulos gerados.

Abrir Imagem

Esta função dá início a todas as operações fornecidas pela interface, uma vez que ela permite a abertura do arquivo de dados (imagem) para a interface.

Após a abertura do arquivo, alguns dados da imagem são exibidos ao usuário que então pode manipulá-los. Como apresentado no diagrama de resposta ao evento da Figura 23 (página 93), os dados são armazenados temporariamente em uma estrutura para melhorar a eficiência de sua manipulação.

No diagrama da Figura 23, podemos observar dois depósitos de dados. O primeiro, ARQUIVO.IMG, representa o arquivo em disco, ou seja, o arquivo que está sendo aberto pelo usuário. O TEMPORÁRIO representa o armazenamento temporário do conteúdo de ARQUIVO.IMG, para facilitar a manipulação e exibição para o usuário, uma vez que todas as operações realizadas a partir da abertura do arquivo de dados serão sobre o armazenamento temporário.

Segundo a análise essencial, a ativação e a reação deste evento será na forma:

Estímulo: Escolha da opção abrir imagem.

Ações: Ler do arquivo selecionado.

Respostas: Exibição dos conteúdos da imagem na interface e armazenamento dessas informações em uma estrutura.

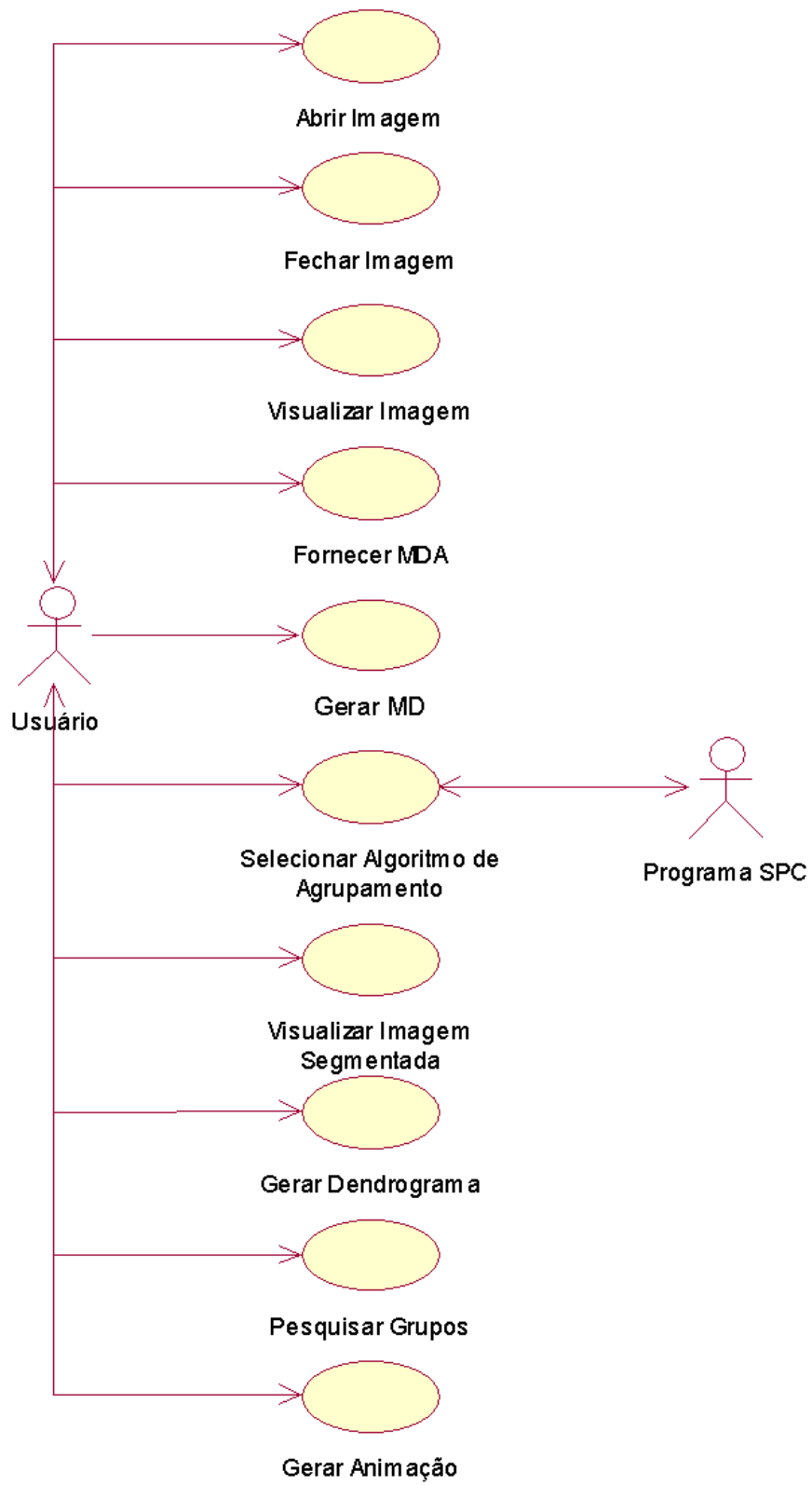


Figura 22: Caso de uso detalhado do sistema.

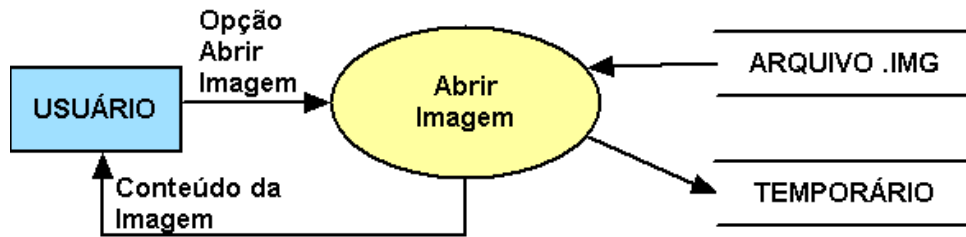


Figura 23: Diagrama de Resposta ao Evento “Abrir Imagem”.

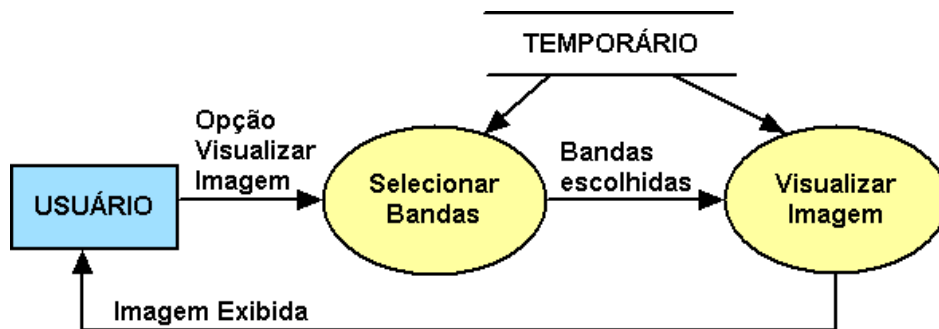


Figura 24: Diagrama de Resposta ao Evento “Visualizar Imagem”.

Fechar Imagem

Esta função é responsável pelo fechamento do arquivo ativo e pela liberação de memória das unidades logicamente alocadas pelo sistema. A ativação e a reação deste evento será na forma:

Estímulo: Escolha da opção fechar imagem.

Ações: Retirar os conteúdos da imagem exibidos na interface, desabilitar os eventos e liberar da memória as unidades logicamente alocadas pelo sistema.

Respostas: Desabilitação das funções principais e informações exibidas na interface, e liberar memória alocada.

Visualizar Imagem

A interface fornece uma exibição simples e pequena da imagem. Para ter uma visão mais detalhada, existe a opção “Visualizar Imagem”. Esta função permite uma visualização da imagem de forma colorida ou em níveis de cinza. A exibição dependerá das bandas escolhidas para cada indicação das componentes do sistema RGB. A Figura 24 ilustra esta operação.

No evento, a ativação e reação são realizadas da seguinte forma:

Estímulo: Escolha da opção visualizar imagem.

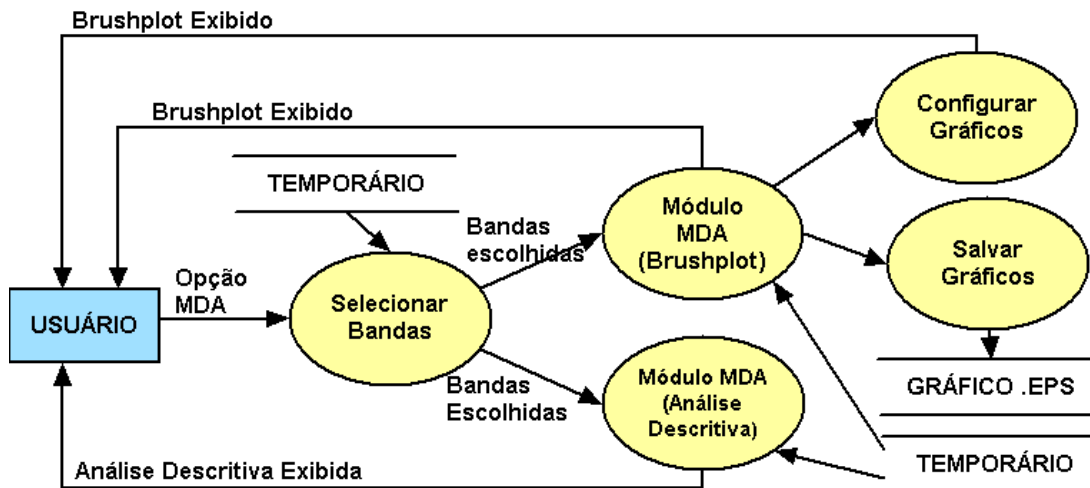


Figura 25: Diagrama de Resposta ao Evento “Fornecer MDA”.

Ações: Obter as bandas especificadas em cada componente do sistema RGB. Compor a imagem pelas bandas selecionadas.

Respostas: Exibição da imagem.

Fornecer MDA

MDA significa “Análise de Dados Multivariados” e esta função faz uma ligação entre o módulo principal e o módulo de análise multivariada que compreende duas etapas: *brushplot* e análise descritiva. A primeira etapa do módulo MDA disponibiliza funcionalidades como visualização dos dados através de histogramas e *scatterplots*. A segunda etapa faz uma análise descritiva dos dados.

O diagrama de resposta ao evento “Fornecer MDA” é apresentado na Figura 25 (página 94). Esta é uma função simples que realiza o armazenamento das bandas que serão utilizadas na análise dos dados e cria o módulo MDA em função das opções escolhidas. Ele retorna ao usuário o acesso ao módulo de análise multivariada, que contém funções específicas do gráfico exibido. A Tabela 30 (página 95) exibe um resumo dessas funções.

No evento, a ativação e reação são realizadas da seguinte forma:

Estímulo: Escolha da opção fornecer MDA.

Ações: Construção da análise multivariada.

Respostas: Exibição da análise multivariada.

Função	Descrição
Selecionar Bandas	Seleção das variáveis que farão parte dos gráficos
Configurar Gráficos	Configuração do modo de exibição do gráfico, por exemplo um scatterplot pode ter seus dados logaritmicamente transformados para uma melhor visualização. Definida na opção <i>Brushplot</i>
Salvar Gráfico	Salva o gráfico corrente em um arquivo com formato .EPS. Definida na opção <i>Brushplot</i>
Análise Descritiva	Exibe valores estatísticos calculados com base no conjunto de dados

Tabela 30: Sub-funções da função “Fornecer MDA”.

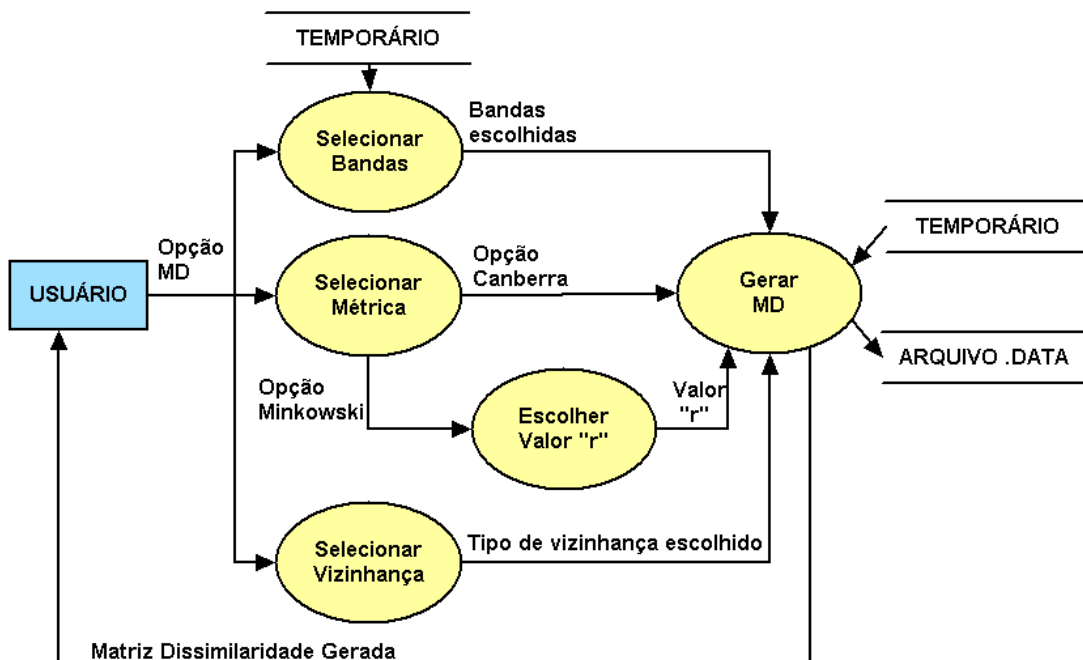


Figura 26: Diagrama de Resposta ao Evento “Gerar MD”.

Gerar MD

A partir desta função é possível gerar a matriz de dissimilaridade. Algumas informações a serem utilizadas devem ser definidas no cálculo da matriz, como as bandas, a métrica e o critério de vizinhança. A saída da função é um arquivo com extensão .DATA, como o exemplo na Tabela 25 (página 85). A Seção 5.2.1 explica a estrutura deste arquivo. O usuário é informado pela interface quando o processo termina. O diagrama de resposta ao evento é apresentado na Figura 26.

Para calcular uma matriz de dissimilaridade, duas opções de métricas são oferecidas, são elas: Canberra e Minkowski (ver Seção 2.3). Se Canberra for escolhida, então a matriz de dissimilaridade será gerada imediatamente após a escolha. Caso Minkowski seja escolhida, é necessária a seleção de um valor para o parâmetro “ r ”. Este parâmetro é utilizado na equação da distância de Minkowski e especifica a equação

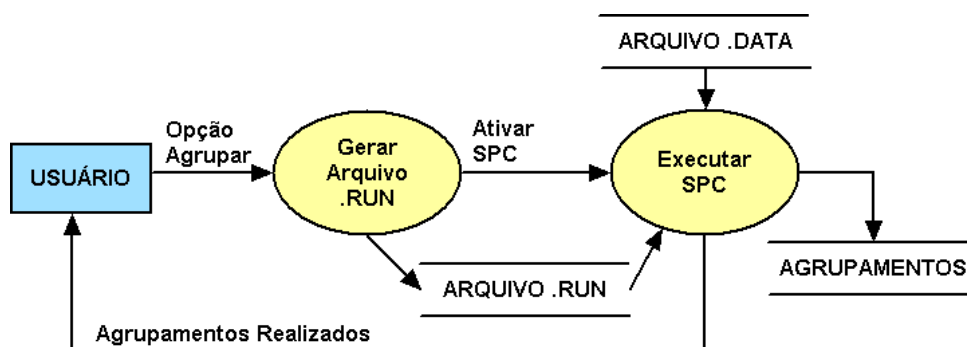


Figura 27: Diagrama de Resposta ao Evento “Selecionar Algoritmo Agrupamento”.

da distância a ser utilizada para a obtenção das dissimilaridades.

Como explicado na Seção 5.2.1, para aplicar a segmentação de imagens ao programa foi inserido o critério de vizinhança no cálculo da dissimilaridade. Os critérios de vizinhança disponibilizados são vizinhança-4 e vizinhança-8.

O evento realiza a ativação e reação da seguinte forma:

Estímulo: Escolha da opção gerar MD.

Ações: Construção da matriz de dissimilaridade a partir das informações obtidas.

Respostas: Armazenamento da matriz em arquivo e mensagem de término do procedimento ao usuário.

Selecionar Algoritmo Agrupamento

Este evento tem a função fornecer a seleção do algoritmo de agrupamento hierárquico a ser utilizado e, após esta escolha, executá-lo. Este fato o torna extensível a outros algoritmos de agrupamento hierárquico. Desde que estes algoritmos possuam as entradas e saídas apresentadas nas Seções 5.2.1 e 5.2.2. Porém, para esta dissertação, só foi aplicado o programa SPC e as explicações serão feitas de acordo com o que foi implementado nela.

O diagrama da Figura 27 mostra que a função para gerar agrupamentos envolve outras duas funções: a de geração do arquivo de informação com extensão .RUN e a execução do SPC.

A primeira função, “Gerar Arquivo .run”, recebe dados do usuário (ver Tabela 16, página 77) para montar o arquivo de informações, que é armazenado fisicamente em disco e servirá como parâmetro para o SPC. Após a criação do arquivo .RUN, a função chama o procedimento que executa o SPC, que se encarrega de ler as informações armazenadas no .RUN e ler a matriz de dissimilaridade armazenada no arquivo .DATA.

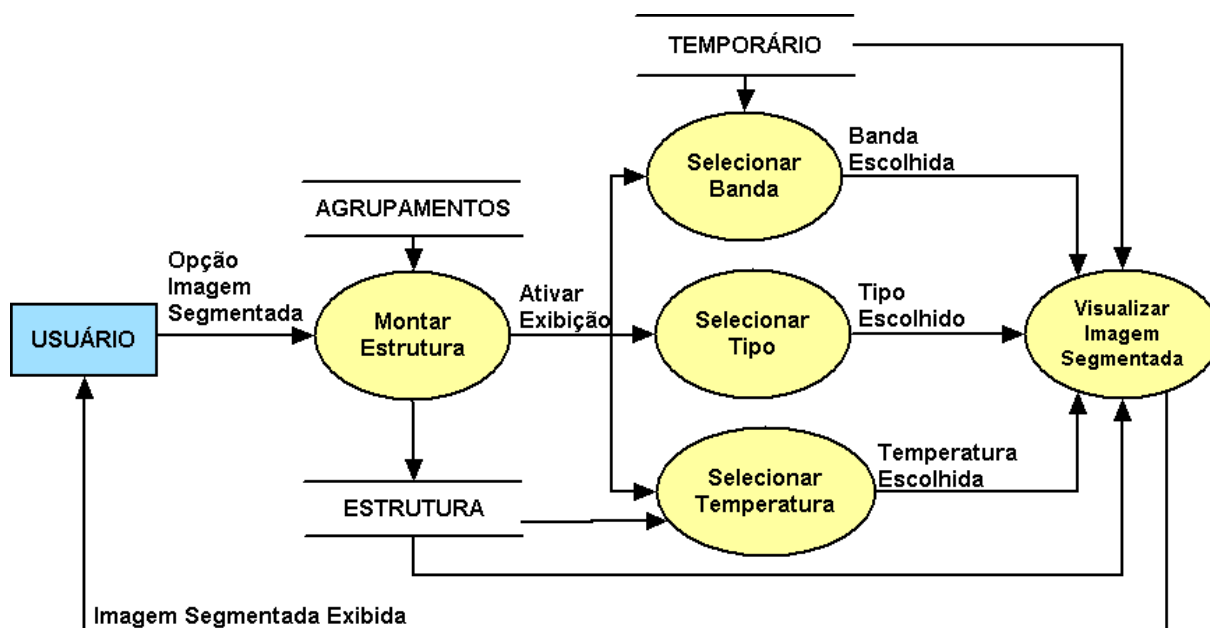


Figura 28: Diagrama de Resposta ao Evento “Visualizar Imagem Segmentada”.

Os arquivos de saída do SPC estão representados pelo arquivo de saída, chamado de AGRUPAMENTOS do diagrama ilustrado na Figura 27 (página 96). Estes arquivos são explicados na Seção 5.2.2. Uma mensagem é mostrada quando a execução do programa é encerrada.

O evento realiza a ativação e reação da seguinte forma:

Estímulo: Escolha da opção selecionar algoritmo de agrupamento.

Ações: Gerar o arquivo .RUN (entrada do programa) e executar o SPC.

Respostas: Arquivos de saída e mensagem de término do procedimento ao usuário.

Visualizar Imagem Segmentada

A imagem segmentada pode ser visualizada por esta função. Para acessá-la, o usuário deve escolher o arquivo a ser utilizado. Os arquivos a serem escolhidos são os arquivos de saída gerados pelo SPC: .DG.01 e .LAB (ver Seção 5.1).

Como ilustrado no diagrama da Figura 28. Quando o usuário escolhe os arquivos, representados por AGRUPAMENTOS, há um procedimento que lê estes arquivos e monta uma estrutura para manipulação dos agrupamentos gerados. A estrutura é um armazenamento temporário para facilitar a manipulação dos agrupamentos a qualquer temperatura e grupo, além do acesso se tornar mais rápido. Esta estrutura é montada para todas as outras opções de visualização dos resultados.

Após a montagem da estrutura, algumas informações devem ser definidas: banda,

tipo de exibição e temperatura. A banda e a temperatura definem qual a banda exibida e qual hierarquia de agrupamento mostrar. Existem dois tipos de exibição da imagem: completa e por contorno. A exibição completa define uma cor a cada segmento. Já a exibição por contorno define uma cor para os contornos de cada segmento.

O procedimento de visualizar imagens une todas as informações obtidas, com as informações armazenadas em `ESTRUTURA` e os dados da imagem em `TEMPORÁRIO` para exibir a imagem segmentada. No evento, a ativação e reação são realizadas da seguinte forma:

Estímulo: Escolha da opção visualizar imagem segmentada.

Ações: Montar estrutura e obter banda, tipo e temperatura. Compor a imagem segmentada.

Respostas: Exibição da imagem segmentada.

Gerar Dendrograma

A função de Gerar Dendrograma fornece o módulo “Dendrogramas”. Este módulo exibe os agrupamentos realizados pelo SPC sob a forma de dendrogramas (ver o Capítulo 2). As funções principais deste módulo estão descritas na Tabela 31, enquanto o diagrama na Figura 29 (página 99) apresenta a estrutura básica deste evento.

Função	Descrição
Montar estrutura	Lê os arquivos de saída do SPC e monta a estrutura a ser utilizada na construção dos dendrogramas
Gerar Dendrograma	A exibição do dendrograma pode ser feita de três maneiras: completa, por temperatura e por indivíduo
Ativar Sensibilidade do Gráfico	Ativa a sensibilidade do gráfico ao clique do mouse, permitindo ao usuário a interação com o dendrograma
Salvar Gráfico	Salva o gráfico corrente em um arquivo com formato .EPS

Tabela 31: Sub-funções da função “Gerar Dendrogramas”.

A sub-função “Gerar Dendrograma” controla a exibição do dendrograma ao usuário. Isto pode ser feito de três maneiras:

completa: esta forma apresentará todo o dendrograma, sem restrições.

por temperatura: nesta forma, temperaturas devem ser escolhidas e apenas os agrupamentos realizados nessas temperaturas são exibidos no gráfico. As temperaturas são escolhidas através de componentes gráficas.

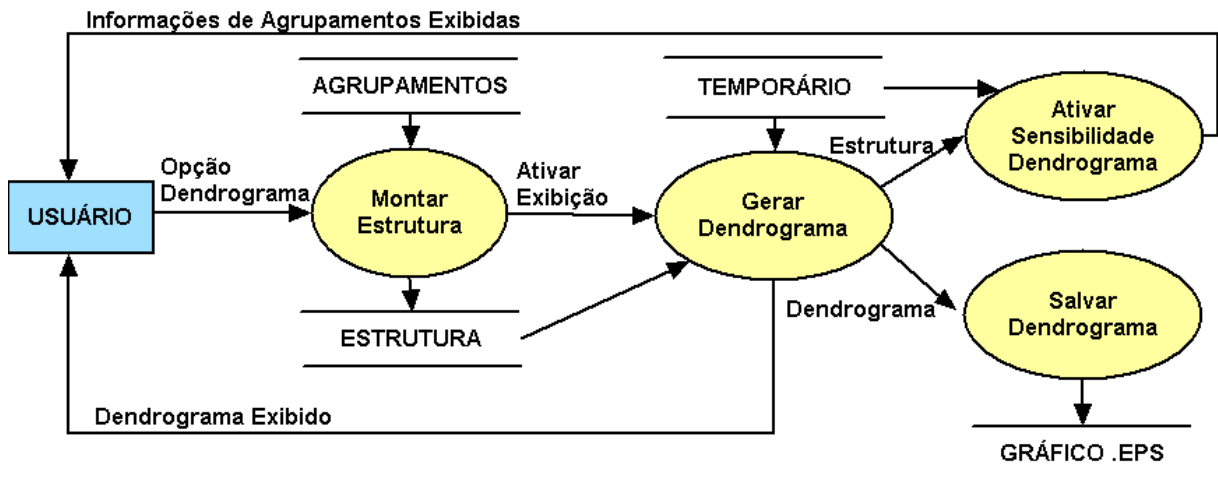


Figura 29: Diagrama de Resposta ao Evento “Gerar Dendrograma”.

por regiões de interesse (ROIs): semelhantemente à forma por temperatura, as regiões devem ser criadas através de um recurso de edição gráfica. O dendrograma exibido tem a mesma estrutura do completo, mas apresenta destaques para os agrupamentos que contêm as ROIs.

Já a sub-função “Ativar Sensibilidade do Gráfico” é uma das funções mais interessantes disponibilizadas na interface, pois ela permitirá que o usuário interaja com o dendrograma e, conseqüentemente, com os agrupamentos nele exibidos.

O objetivo dessa sub-função é fazer o usuário selecionar algum grupo em qualquer parte do dendrograma. O padrão é o ponteiro do mouse apresentar a forma de uma seta. Mas, quando este ponteiro está posicionado nas regiões do gráfico que indicam os grupos, ele muda para a forma de cruz. Esta mudança indicará ao usuário que naquela região, sob as condições de temperatura exibidas no dendrograma, aconteceu um agrupamento. O usuário pode ainda clicar nessa região do gráfico, onde o ponteiro está sob a forma de cruz, e obter informações sobre a quantidade de indivíduos contidos naquele agrupamento e quais são aqueles indivíduos, além de uma análise estatística envolvendo os dados e as variáveis do conjunto de dados. Essa função é bastante útil em situações onde o conjunto de dados utilizado no agrupamento é muito grande, impossibilitando uma visão apropriada das informações contidas no dendrograma. Uma ilustração de todo este processo é exibida na seção posterior.

No evento, a ativação e reação são realizadas da seguinte forma:

Estímulo: Escolha da opção gerar dendrogramas.

Ações: Montar a estrutura dos agrupamentos a ser utilizada, obter o tipo de dendrograma selecionado e construí-lo.

Respostas: Exibição do dendrograma ou salvá-lo em arquivo ou obtenção de infor-

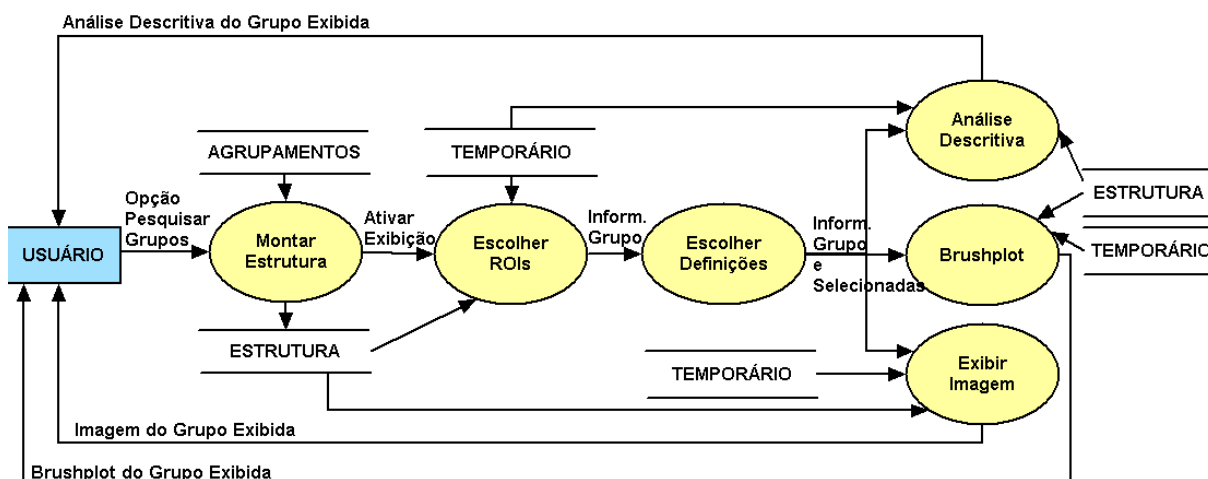


Figura 30: Diagrama de Resposta ao Evento “Procurar Grupo”.

mações sobre os grupos ilustrados no dendrograma.

Procurar Grupo

Esta função localiza e exibe os agrupamentos que contêm todas as regiões definidas na pesquisa. É apenas mais uma opção para localização de grupos, podendo estes serem visualizados através de imagens ou análises multivariadas.

A Figura 30 exibe o diagrama de resposta a este evento. Inicialmente, os arquivos contendo os agrupamentos devem ser lidos, organizados e armazenados em uma estrutura. Depois, escolhe-se as regiões de interesse (ROIs) e faz-se uma busca aos grupos para cada temperatura. As temperaturas que tiverem agrupamentos com as ROIs definidas são informadas. Em seguida, algumas informações como banda a ser visualizada, tipo de exibição (por contorno ou completa) e temperatura a serem exibidos os grupos são definidas. Após estas escolhas, o usuário pode selecionar qual resultado quer visualizar: a imagem contendo os grupos pesquisados ou a análise multivariada. As exibições dos resultados são realizadas de forma independente, podendo os três serem analisados ao mesmo tempo.

O evento realiza a ativação e reação da seguinte forma:

Estímulo: Escolha da opção procurar grupo.

Ações: Montar a estrutura dos agrupamentos a ser utilizada, definir ROIs e informações de exibição dos resultados e realizar a busca dos grupos.

Respostas: Exibição imagens e análises multivariadas.

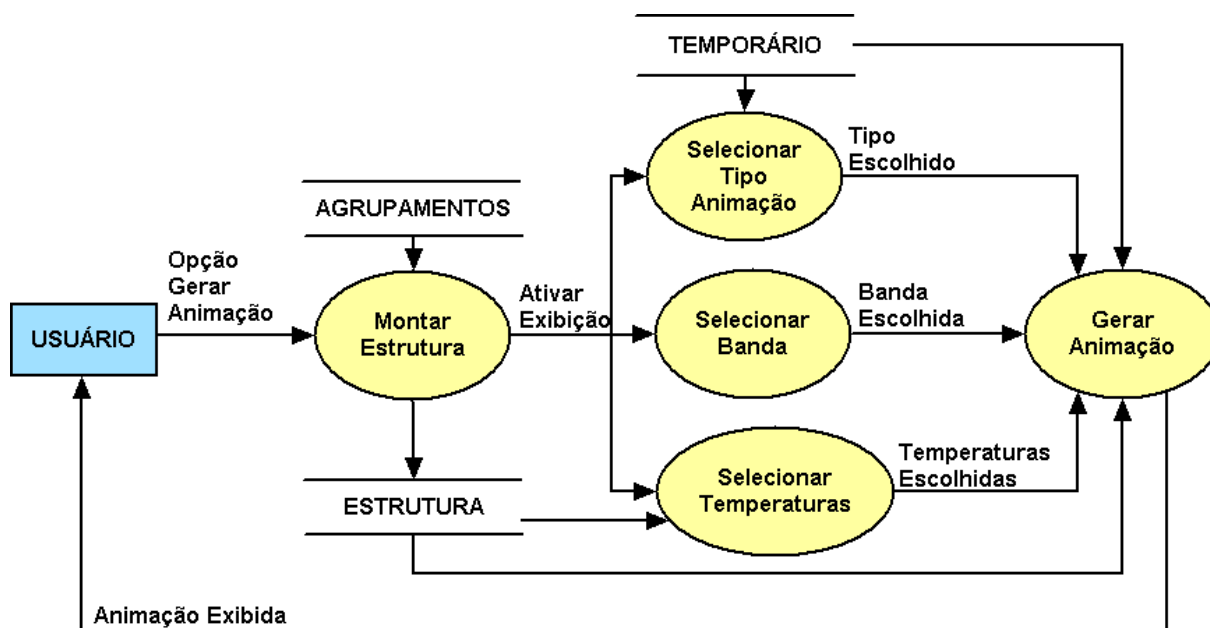


Figura 31: Diagrama de Resposta ao Evento “Gerar Animações”.

Gerar Animações

Este evento é uma forma inovadora e intuitiva de se entender os agrupamentos realizados pelo SPC. Ele provê animações com os grupos formados através de duas possibilidades: segmentos formados a cada temperatura ou imagens segmentadas diante da hierarquia estabelecida pelo programa.

O diagrama da Figura 31 ilustra a estrutura base deste evento. Como os outros eventos de exibição dos resultados, ele necessita que os arquivos sejam disposto na forma da estrutura estabelecida em ESTRUTURA. Com a estrutura montada, deve-se escolher as informações necessárias à construção da animação: o tipo de animação a ser feita, qual banda exibir e as temperaturas a serem utilizadas na animação. A quantidade de temperatura dependerá do tipo de animação escolhido.

Definidas as informações básicas, pode-se criar as animações que podem ser subdivididas em duas funções ilustradas na Tabela 32.

Função	Descrição
Grupos por temperatura	Exibe a animação de todos os segmentos presentes numa temperatura. Neste caso, o programa só aceita a seleção de uma temperatura
Temperaturas por algoritmo	Exibe a animação das temperaturas selecionadas para destacar o nível de hierarquia estabelecido

Tabela 32: Sub-funções da função “Gerar Animações”.

O evento realiza a ativação e reação da seguinte forma:

Estímulo: Escolha da opção gerar animação.

Ações: Montar a estrutura dos agrupamentos a ser utilizada, definir as informações necessárias à construção, construir a animação.

Respostas: Exibição das animações.

Nesta seção, pode-se ter uma análise do funcionamento da interface. A próxima seção ilustra a interface implementada.

5.3.2 Implementação da Interface

A seção anterior analisou e descreveu de uma maneira textual o funcionamento da interface proposta. Nesta seção será exibida essa interface, suas funções correspondentes e as implementações realizadas.

Módulo Principal

O módulo principal da interface, exibido na Figura 32 (página 103), é responsável por todas as funções ilustradas na Figura 22 (página 92). Estas funções podem ser acessadas através de menus e botões para facilitar a interação com o usuário. A seguir as principais opções fornecidas por este módulo são explicadas.

Menu *File*: Este menu apresenta três opções como ilustra a Figura 33 (página 103).

A opção *Open* realiza o evento “Abrir imagem” que exibe tanto uma representação da imagem quanto suas informações na interface. A opção *Close* realiza o evento “Fechar Imagem”, retirando as informações exibidas na interface e liberando as unidades lógicas ocupadas pela imagem. Estas duas opções podem ser acessadas por seus respectivos botões na interface. A última opção deste menu é a *Exit*, que fecha todo o programa.

Menu *Options*: É composto por oito opções como ilustra a Figura 34 (página 104).

Todas essas opções são também disponibilizadas através de botões que são detalhados mais adiante.

Opção *Open*: Esta opção corresponde ao evento de “Abrir Imagem” explicado na Seção 5.3.1. Ela pode ser acessada tanto pelo menu *File* quanto pelo botão padronizado de abrir arquivos. Quando esta opção é ativada, uma versão minimizada da imagem é exibida, assim como algumas de suas informações. Além disso, as opções principais da interface são habilitadas. A Figura 32 (página 103) ilustra estas respostas do evento.

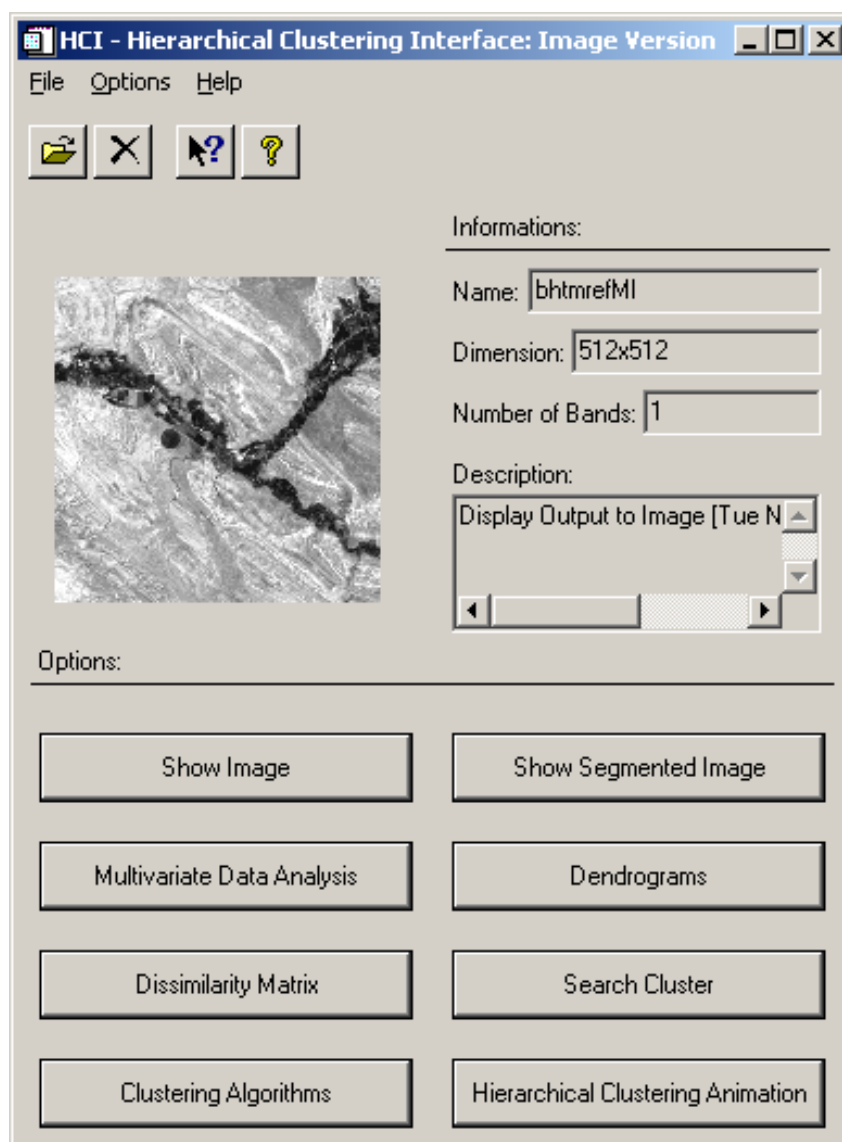
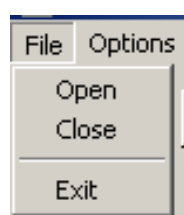


Figura 32: Tela Principal da Interface.

Figura 33: Opção *File* do Menu Principal.

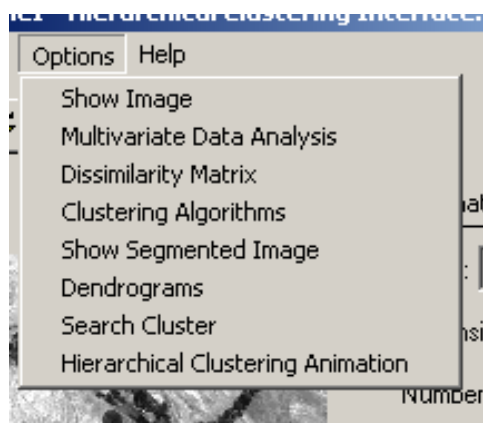


Figura 34: Opção *Options* do Menu Principal.

Opção *Close*: Esta opção corresponde ao evento de “Fechar Imagem” explicado na Seção 5.3.1. Ela pode ser acessada tanto pelo menu *File* quanto pelo botão de exclusão (ver Figura 32, página 103). Quando esta opção é ativada, os dados da imagem são retirados da interface e as opções principais da interface são desabilitadas. A Figura 35 (página 105) ilustra estas respostas do evento.

Opção *Show Image*: Esta opção corresponde ao evento de “Visualizar Imagem” explicado na Seção 5.3.1. Ela pode ser acessada tanto pelo menu *Options* quanto pelo botão na janela principal (ver Figura 32, página 103). Esta opção permite ao usuário visualizar a imagem de forma colorida ou em níveis de cinza dependendo das bandas selecionadas a cada componente do sistema RGB. Este tipo de exibição foi baseado no software ENVI (ver Seção 1.3.2), onde na análise, percebeu-se que as opções disponíveis na apresentação da imagem (colorida ou em níveis de cinza) pode ser simplificada. Esta redução de informações foi aplicada à janela considerada principal deste evento (Figura 36(a), página 105). Inicialmente, as bandas devem ser selecionadas pela janela principal no evento. Após a ativação do botão *Show*, a janela secundária (Figura 36(b)) é disponibilizada. Diante de pesquisas realizadas nas componentes gráficas apresentadas pelo IDL, chegou-se a janela secundária apresentada na Figura 36(b). Esta janela é bem flexível, fornecendo *zoom*, cópia da imagem, impressão, além de edição através de linhas, retângulos e textos. Há um controle para fechar as janelas, onde a janela secundária é eliminada quando a principal é fechada.

Opção *Multivariate Data Analysis*: Esta opção realiza a ligação entre o módulo principal e o módulo MDA. Ela pode ser acessada tanto pelo menu *Options* quanto pelo botão na janela principal (ver Figura 32). O módulo MDA está descrito nesta seção.

Opção *Dissimilarity Matrix*: Esta opção corresponde ao evento de “Gerar MD” explicado na Seção 5.3.1. Ela tem como objetivo construir a matriz de dissimilaridade. O acesso a este evento pode ser realizado tanto pelo menu *Options* quanto

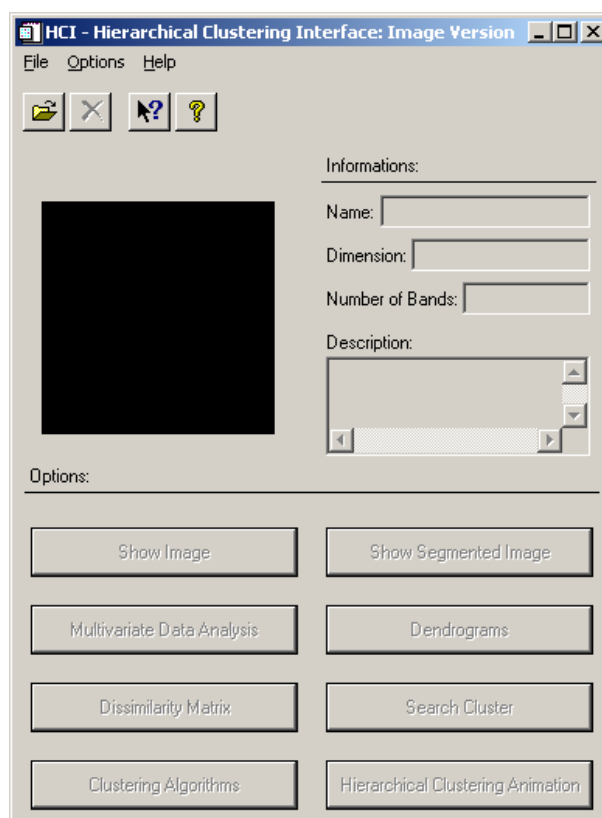
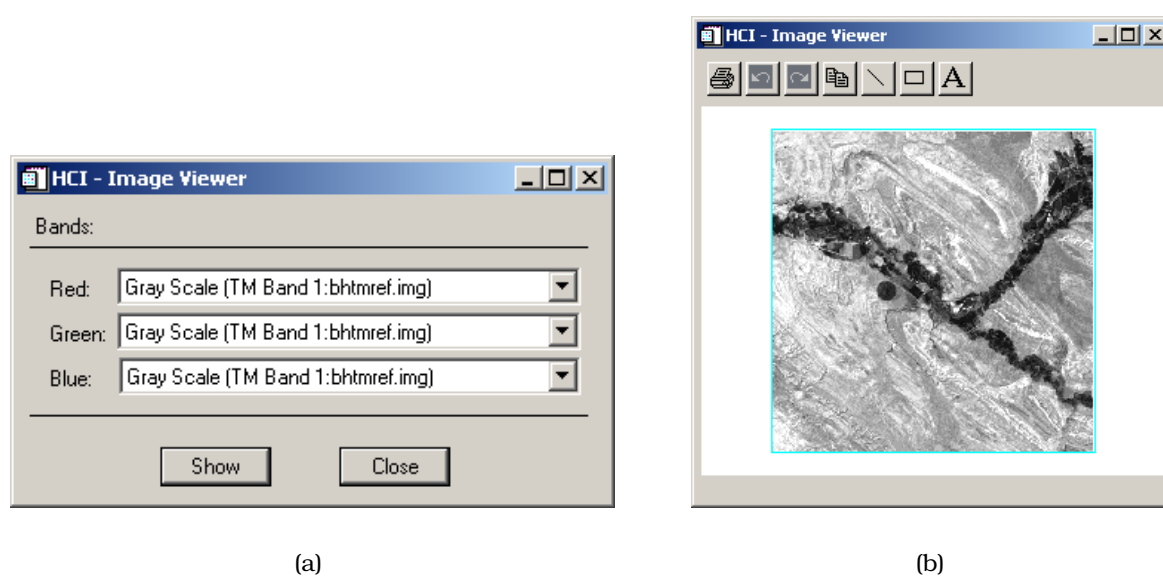


Figura 35: Resultados da Opção *Close* do Módulo Principal.



(a)

(b)

Figura 36: Opção *Show Image* do Módulo Principal: (a) janela principal e (b) janela secundária.

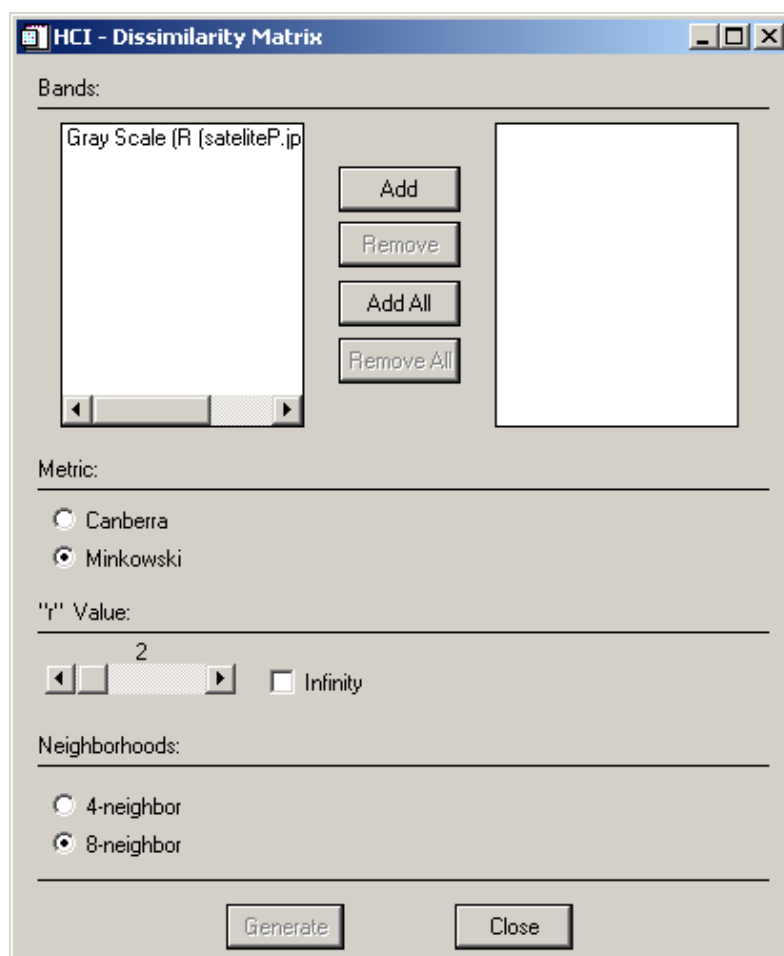


Figura 37: Opção *Dissimilarity Matrix* do Módulo Principal.

pelo botão na janela principal (ver Figura 32, página 103). Quando isto ocorre, a janela da Figura 37 é aberta.

Para a construção da matriz de dissimilaridade algumas informações devem ser definidas. A escolha das bandas informam os atributos a serem calculados pela métrica escolhida. O usuário pode escolher entre duas métricas: Minkowski e Camberra (ver Seção 2.3). Se a opção Minkowski for a escolhida, opção “*r*” Value na janela é ativada para que o usuário determine o valor do parâmetro *r*. Este parâmetro especifica a equação da distância a ser utilizada para a obtenção das dissimilaridades. Alguns casos especiais são a distância Euclidiana ($r = 2$) e distância Manhattan ($r = 1$). Se a opção Camberra for escolhida, opção “*r*” Value na janela é desabilitada. Além disso, para aplicar a segmentação de imagens ao SPC (ver Seção 5.2.1) há a necessidade de se utilizar um critério de vizinhança. A interface disponibiliza a vizinhança-4 e vizinhança-8.

Após definidas as opções disponibilizadas pela janela, pode-se clicar no botão *generate* para criar o arquivo contendo a matriz de dissimilaridade. Este arquivo de entrada do SPC é criado após escolha do seu nome e indicação de caminho. Com o término deste processamento, uma mensagem é exibida ao usuário.

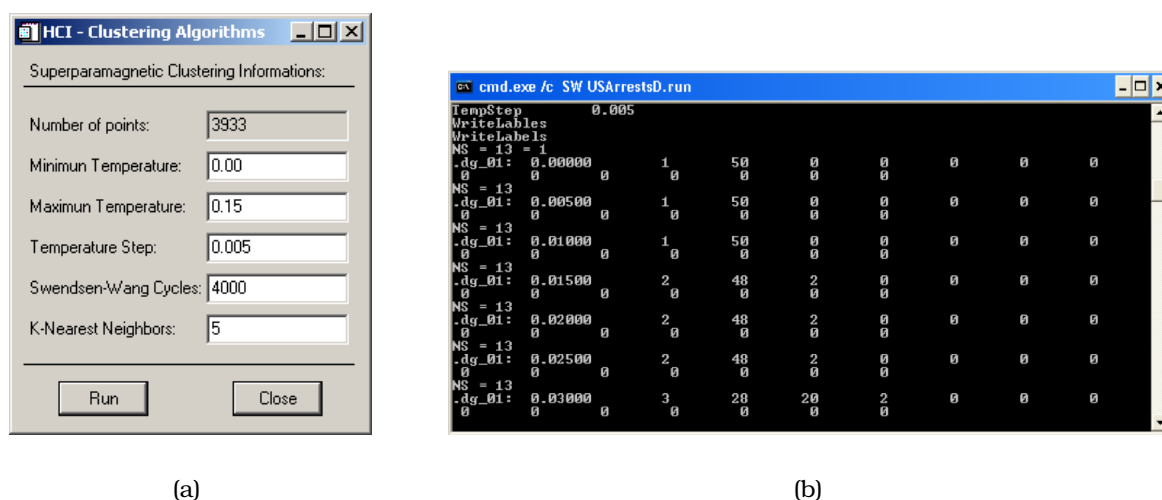


Figura 38: Opção *Clustering Algorithms* do Módulo Principal: (a) janela principal e (b) janela de execução do SPC.

Opção *Clustering Algorithms*: Realiza a ligação entre o módulo principal e o programa SPC. Esta opção corresponde ao evento “Selecionar Algoritmo Agrupamento” (ver Seção 5.3.1). Ela pode ser acessada tanto pelo menu *Options* quanto pelo botão na janela principal (ver Figura 32, página 103). Quando esta opção é acessada, uma janela é disponibilizada (Figura 38(a)) para definir os parâmetros a serem armazenados no arquivo `.RUN`. Após acionar o botão *Run*, é solicitado ao usuário que escolha um arquivo com extensão `.DATA`. Este arquivo deve conter uma matriz de dissimilaridade válida. A janela da Figura 38(b) é exibida em seguida. Ela mostra a chamada feita ao programa SPC e os seus resultados, ou seja, os agrupamentos sendo realizados.

Quando o usuário aciona o botão *Run*, o sistema internamente chama a execução do SPC através do comando abaixo. Este código no IDL integra um executável ao sistema desenvolvido:

```
spawn , 'SW ' + file,
```

onde *SW* é o nome do executável do programa SPC e *file* é o nome do arquivo `.RUN` que será executado. O procedimento do IDL, *spawn*, executa um comando ou uma série de comandos e tem como opção a possibilidade de exibir uma janela *shell* ou não. No exemplo apresentado o *shell* foi exibido para ilustração.

Opção *Show Segmented Image*: Esta opção realiza a ligação entre o módulo principal e o módulo imagem segmentada. Ela pode ser acessada tanto pelo menu *Options* quanto pelo botão na janela principal (ver Figura 32, página 103). O módulo imagem segmentado é apresentado nesta seção.

Opção *Dendrograms*: Esta opção realiza a ligação entre o módulo principal e o módulo dendrograma, que será detalhado mais adiante. Ela pode ser acessada

tanto pelo menu *Options* quanto pelo botão na janela principal (ver Figura 32, página 103).

Opção *Search Cluster*: Esta opção pode ser executada tanto pelo menu *Options* quanto pelo botão na janela principal (ver Figura 32). Ela realiza a ligação entre o módulo principal e o módulo procurar grupo, que é detalhado mais adiante.

Opção *Hierarchical Clustering Animation*: Esta opção pode ser executada tanto pelo menu *Options* quanto pelo botão na janela principal (ver Figura 32). Ela realiza a ligação entre o módulo principal e o módulo animação, cujas descrições são apresentadas mais adiante.

As próximas subseções apresentam os outros cinco módulos do sistema, são eles: MDA, imagem segmentada, dendrogramas, procurar grupos e o animação.

Módulo MDA

A Figura 39(a) (página 111) apresenta a tela principal do módulo MDA. Este módulo corresponde ao evento “Fornecer MDA” (ver Seção 5.3.1) e realiza uma análise multivariada dos dados presentes na imagem através de duas formas: *brushplot* e análise descritiva. Os botões *Brushplot* e *Descriptive Statistics* só são habilitados quando pelo menos uma banda é selecionada.

Quando ativado o botão *Brushplot* a janela da Figura 40 (página 112) é exibida. O gráfico apresentado nesta janela é um *brushplot* em sua forma triangular. Este apresenta em sua diagonal principal os histogramas correspondentes às bandas selecionadas na Figura 39(a). Nas outras posições do *brushplot* são apresentados os *scatterplots*, que são gráficos de correlação entre as bandas selecionadas.

São disponibilizadas também opções de configuração para esses gráficos. Para os histogramas é possível configurar o tipo e o valor do *bin* a ser apresentado; o tipo e a largura da linha do gráfico (pontilhada, tracejada, etc.); o tipo, a largura e o tamanho da fonte; e o tipo e a largura dos eixos dos histogramas (ver Figura 39(b), página 111). Para os *scatterplots* também é possível configurar o tipo e a largura dos eixos; o tipo, a largura e o tamanho da fonte; o tipo e o tamanho do símbolo a ser exibido nos gráficos, tais como ponto, asterisco, sinal de somar, diamante etc.; e o tipo da variável, na forma normal ou logarítmica (ver Figura 39(c)).

Outra opção disponibilizada é a de análise descritiva (ver Figura 41 na página 113). Através dessa opção são exibidas várias quantidades estatísticas para as bandas selecionadas na janela principal deste módulo. Essas quantidades estão descritas a seguir:

Média Aritmética: É uma medida de localização que informa o valor central da amostra e obtém-se a partir da seguinte expressão: $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$, onde x_i repre-

senta o elemento da amostra e n a sua dimensão.

MAD: Define-se desvio médio absoluto (MAD) para uma série de n elementos da amostra como sendo a média aritmética simples dos módulos dos desvios desses n elementos. É dado por: $MAD = \frac{1}{n} \sum_{i=1}^n |x_i - \bar{x}|$.

Mediana: Dado que elementos da amostra estejam ordenados, a mediana é o valor (pertencente ou não à amostra) que a divide ao meio, isto é, 50% dos elementos da amostra são menores ou iguais à mediana e os outros 50% são maiores ou iguais à mediana. Dada a notação $x_{1:n}, x_{2:n}, \dots, x_{n:n}$ para os elementos da amostra ordenada, tem-se a seguinte expressão para o cálculo da mediana:

$$mediana = \begin{cases} x_{\frac{n+1}{2}:n} & \text{se } n \text{ é ímpar,} \\ \frac{1}{2}(x_{\frac{n}{2}:n} + x_{\frac{n}{2}+1:n}) & \text{se } n \text{ é par.} \end{cases}$$

Máximo e Mínimo: o maior e menor valor, respectivamente, do conjunto de dados.

Variância: é uma medida da variabilidade dos dados em torno da média. É dada por:

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2.$$

Desvio Padrão: é uma medida da variabilidade ou dispersão com as mesmas unidades que os dados. É dada por: $S = \sqrt{S^2}$. Quanto maior for o valor de S , mais dispersos são estes dados. Se $S = 0$, então não existe variabilidade, isto é, os dados são todos iguais.

CV: O fato do desvio padrão ser expresso na mesma unidade dos dados limita o seu emprego quando deseja-se comparar duas ou mais séries de valores, relativamente à sua dispersão ou variabilidade, quando expressas em unidades diferentes. Para contornar esta dificuldade, pode-se caracterizar a dispersão ou variabilidade dos dados em termos relativos a seu valor médio. Esta medida é denominada de CVP (Coeficiente de Variação de Pearson), sendo definida pela expressão: $CV = \frac{S}{\bar{x}}$.

Coeficiente de Assimetria: mede o grau de simetria entre os pontos à esquerda e à direita do ponto central. É obtida a partir do terceiro momento central. É dada por: $Skewness = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{S}\right)^3$.

Curtose: mede o grau de achatamento de uma distribuição. É obtida a partir do quarto momento central. Dada por: $Kurtosis = \frac{1}{n} \sum_{i=1}^n \left[\left(\frac{x_i - \bar{x}}{S}\right)^4 - 3\right]$.

Matriz de correlação: é uma matriz quadrada, simétrica que informa a dispersão entre dados multivariados. Em cada posição (i, j) da matriz, o coeficiente de correlação entre dados bivariados (x_k, y_k) é informado. O coeficiente de correlação entre duas amostras, denominado r , é dado por:

$$r = \frac{S_{XY}}{\sqrt{S_{XX}}\sqrt{S_{YY}}},$$

```

thisDevice = !D.Name

Set_Plot, 'PS'

Device, FileName=file, xsize=8, ysize=8, /inches,
  xoffset=2.25, yoffset=3.5, /ENCAPSULATED

IlustraBrushplot, Nstruct, event.id, val3, 0

Device, /Close_File

Set_Plot, thisDevice

```

Tabela 33: Recurso IDL para salvar imagem com extensão .EPS.

onde $S_{XY} = \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})$. Para as posições localizadas na diagonal da matriz, este valor é 1, visto que será um dado do tipo (x_k, x_k) .

Ainda é disponibilizada na interface uma opção para salvar os *brushplots* gerados no formato .EPS. Quando o usuário aciona o botão *Save*, o sistema internamente executa o procedimento descrito na Tabela 33. Este recurso fornecido pelo IDL facilita a criação de arquivos em .EPS. A primeira linha desse procedimento armazena na variável *thisDevice* o nome do dispositivo ativo. *!D* é um variável do sistema e equivale a uma estrutura que contém informações sobre os gráficos correntes do dispositivo de saída. O seu campo *Name* guarda uma cadeia de caracteres que representa o nome do dispositivo ativo.

O procedimento *Set_Plot*, cuja sintaxe é : `SET_PLOT, Device`, determina o dispositivo de saída usado pelos procedimentos gráficos do IDL. Para salvar no formato .EPS, o dispositivo especificado é o 'PS', representando o formato *PostScript*. O passo seguinte é definir as características do dispositivo 'PS' especificado através do procedimento:

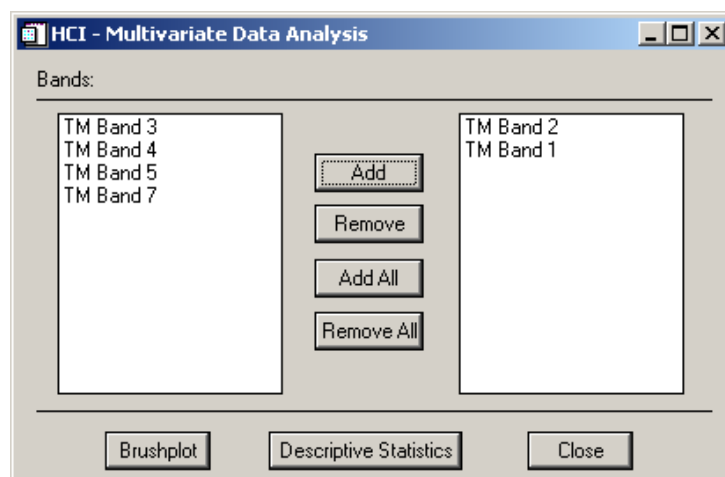
```

Device, FileName=file, xsize=8, ysize=8, /inches, xoffset=2.25 $
  , yoffset=3.5, /ENCAPSULATED

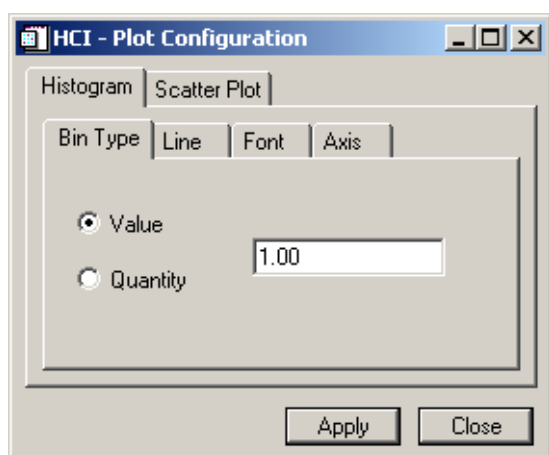
```

Este procedimento define características do dispositivo como o nome do arquivo onde os gráficos serão salvos, o seu tamanho e o seu formato, nesse caso específico *ENCAPSULATED*, para o formato .EPS. Em seguida o *brushplot* é ilustrado, mas agora em outro dispositivo, no caso em um arquivo .EPS e não mais no dispositivo de saída padrão, a tela do computador.

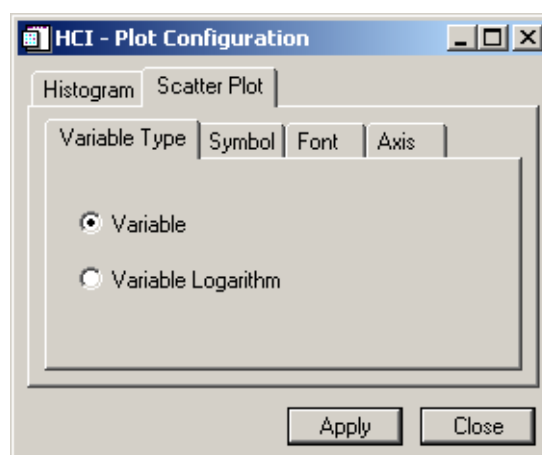
As duas últimas linhas do procedimento retornam o controle para o dispositivo de saída padrão, anteriormente armazenado na variável *thisDevice*.



(a)



(b)



(c)

Figura 39: Módulo MDA: (a) janela principal, (b) janela de configuração dos histogramas e (c) janela de configuração dos *scatterplots*.

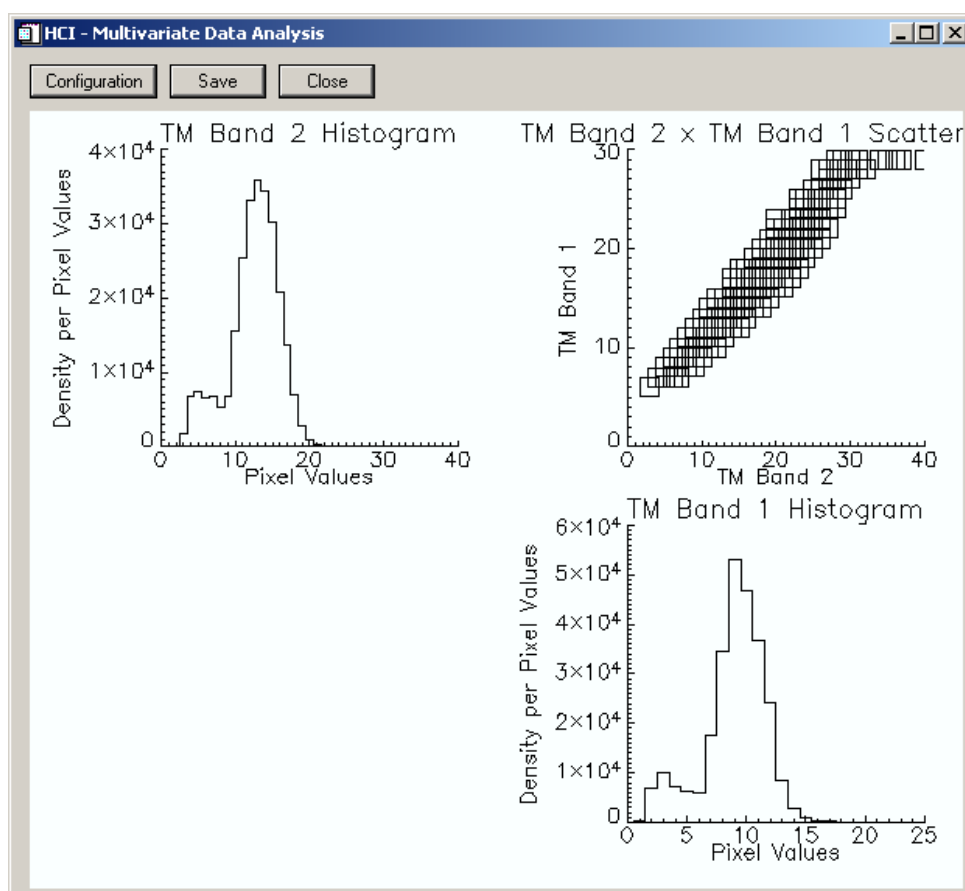


Figura 40: Janela *Brushplot* do Módulo MDA.

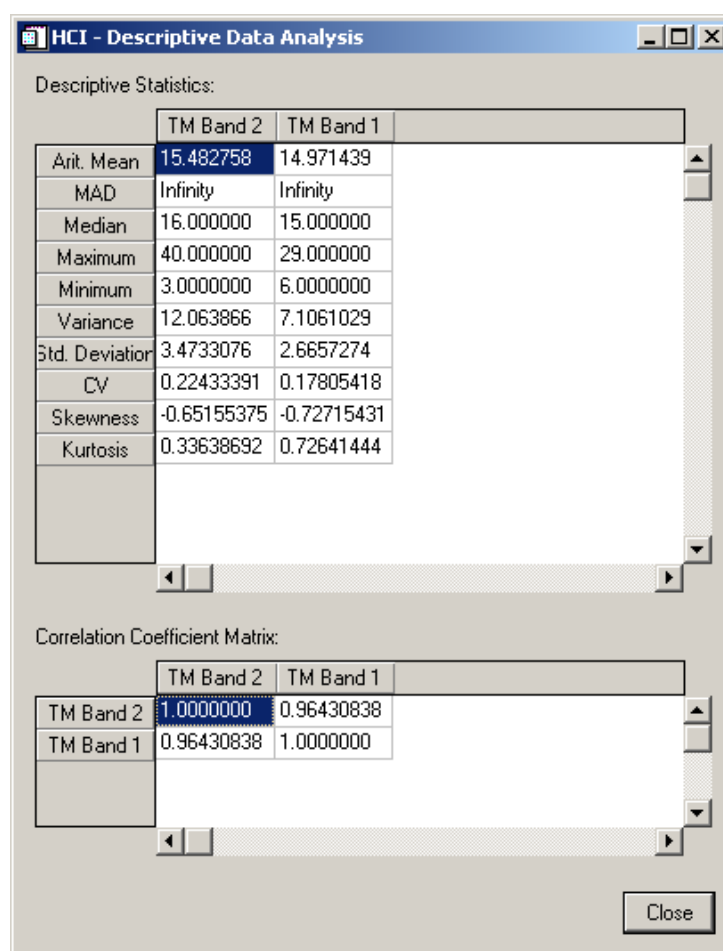


Figura 41: Janela de análise descritiva do Módulo MDA.

Módulo Imagem Segmentada

Este módulo corresponde ao evento “Visualizar Imagem Segmentada” explicado na Seção 5.3.1. Ele exibe os resultados obtidos pelo programa SPC através dos segmentos encontrados nestes resultados e que compõem a imagem. A janela principal deste módulo é a apresentada na Figura 42(a).

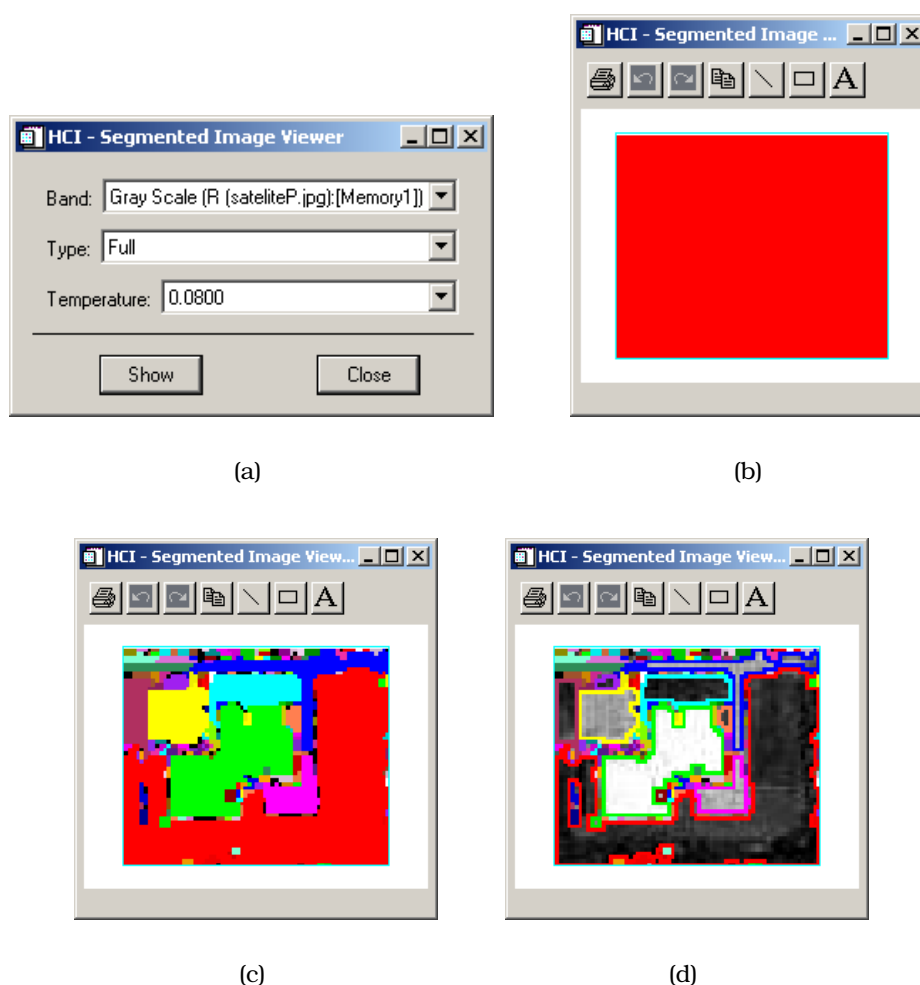


Figura 42: Módulo Imagem Segmentada: (a) janela principal, (b) imagem segmentada com temperatura de valor baixo, (c) janela de exibição com exemplo de segmentos completos e (d) janela de exibição com exemplo de segmentos por contorno.

Para visualizar a imagem segmentada, três opções devem ser selecionadas: a banda a ser exibida, o tipo de apresentação dos segmentos e a temperatura cuja composição define a imagem segmentada. Quando na janela principal a opção *full* em *type* é selecionada, todos os pixels de cada segmento são definidos por uma cor e a junção de todos os segmentos compõem a imagem (ver Figura 42(c)). Quando a opção *contour* é selecionada, só os contornos de um segmento são coloridos como ilustra a Figura 42(d).

Como explicado na Seção 3.6, a baixas temperaturas os resultados tendem a for-

mar um único segmento. A Figura 42(b) apresenta a imagem segmentada com a opção *full* e a temperatura mais baixa, ou seja, de valor 0.0. A medida que a temperatura aumenta, a quantidade de grupos (segmentos) nesta imagem tenderá a aumentar, como é o caso da Figura 42(c).

Módulo Dendrograma

O módulo Dendrograma é representado pela tela da Figura 43 e apresenta suas principais sub-funções na Tabela 31 (página 98). Seu objetivo principal é exibir os resultados obtidos com o programa SPC em forma de **dendrogramas**, ou seja, exibir hierarquicamente os agrupamentos.

Como visto na Seção 2.6, dendrogramas são diagramas bidimensionais que ilustram as fusões ou divisões que acontecem com o conjunto de dados, baseado em alguma técnica. Os dendrogramas desse projeto apresentam no eixo horizontal os indivíduos (pixels) que serão agrupados e no eixo vertical as temperaturas nas quais ocorrerem os agrupamentos.

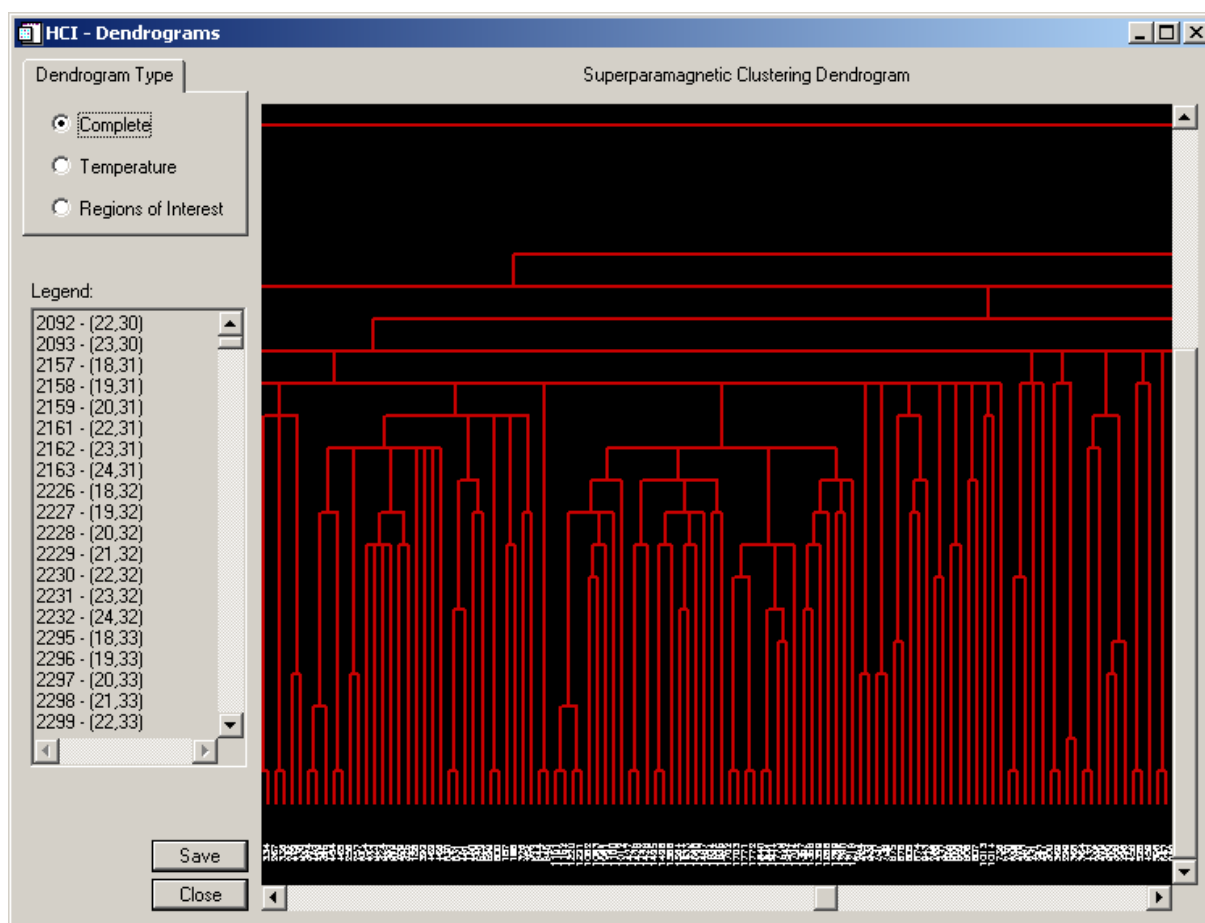


Figura 43: Tela do Módulo Dendrograma.

A legenda apresentada na Figura 43 exibe os indivíduos na ordem em que eles

aparecem no dendrograma. Isto é necessário porque se os verdadeiros nomes dos indivíduos fossem exibidos no gráfico haveria uma grande confusão em qualquer conjunto de dados consideravelmente grande. Para resolver esse problema, índices foram atribuídos aos indivíduos e estes são mostrados no dendrograma.

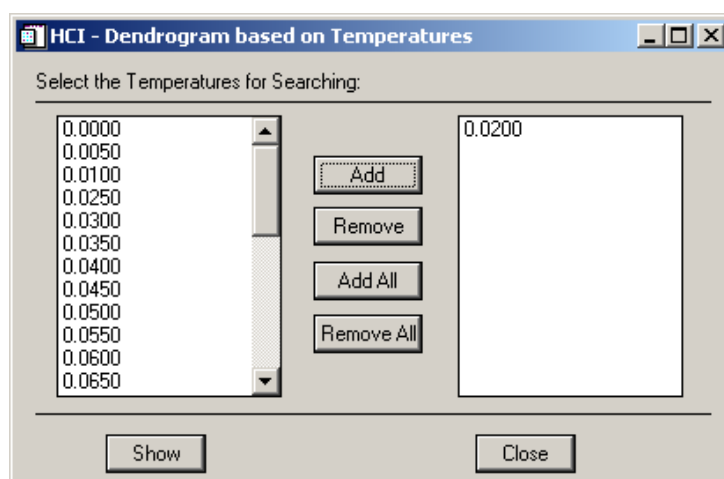
Uma outra característica desse módulo é a disponibilidade de três tipos de opções para a visualização dos dendrogramas, são elas: *Complete*, *Temperature* e *Regions of Interest*. Na opção *Complete* (ver Figura 43, página 115), o dendrograma é completamente exibido sem restrições, ou seja, ele é exibido com todos os agrupamentos e todas as temperaturas nas quais ocorreram agrupamentos.

A opção *Temperature* permite a visualização dos agrupamentos que aconteceram a determinadas temperaturas. Ao escolher esta opção, a janela da Figura 44(a) (página 117) é exibida para que o usuário selecione as temperaturas desejadas para fazerem parte do dendrograma. A opção *Show* desta mesma figura aplica as condições estipuladas, ou seja, constrói o dendrograma na janela da Figura 43 somente com as temperaturas desejadas.

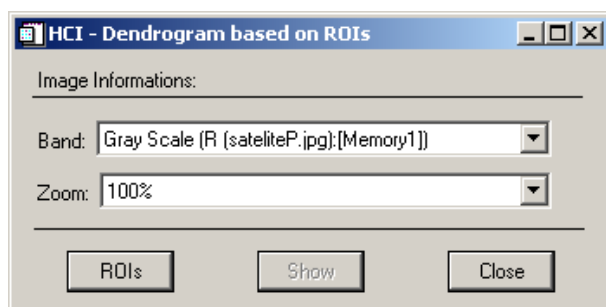
A última opção é a *Regions of Interest* que exibe a janela da Figura 44(b) para que o usuário selecione as regiões de interesse na imagem. Estas regiões também são chamadas de ROIs na interface. Inicialmente, deve-se escolher qual a banda e tamanho da imagem a ser apresentados na seleção. Ao clicar na opção *ROIs*, a janela da Figura 44(c) é exibida e fornece ao usuário várias formas de seleção das ROIs. A opção *Show* controla o dendrograma na janela da Figura 43 em função das regiões selecionadas. Os agrupamentos realizados com estas ROIs são exibidos através da cor amarela.

Este módulo, assim como o módulo MDA, também apresenta uma opção *Save*, onde os dendrogramas são salvos em formato .EPS. O procedimento para salvar esses gráficos é o mesmo apresentado no módulo MDA.

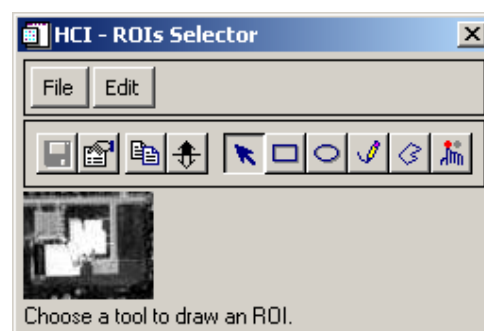
Uma das opções mais interessantes que este módulo possui é a de nó sensível ao clique do mouse. O ponto sensível representado na Figura 45(a) (página 118) é um nó do dendrograma e indica que um agrupamento aconteceu, ou seja, representa um grupo encontrado pelo SPC à temperatura definida naquela altura. Quando um evento de clique do mouse é efetuado na área exibida são mostradas algumas informações sobre este grupo (segmento) na janela da Figura 45(b), são elas: o próprio segmento exibido através da opção *Show Image* e a análise multivariada através das opções *Brushplot* e *Descriptive Statistics*. A opção *Show Image* apresenta este segmento na imagem com as mesmas propriedades do módulo Visualizar Imagem Segmentada. A opção de análise multivariada apresenta a mesma estrutura do módulo MDA descrito nesta seção.



(a)



(b)



(c)

Figura 44: Módulo Dendrograma: (a) janela de *Temperature*, (b) janela principal de ROIs e (c) janela seleção de ROIs.

Módulo Procura Grupo

Este módulo corresponde ao evento “Procurar Grupos” (ver Seção 5.3.1). Ele tem a função de apresentar algumas informações presentes nas temperaturas que tiverem as ROIs selecionadas em um único grupo. Esta opção é uma forma mais prática de se obter um determinado grupo sem sua análise pelos dendrogramas.

Quando este módulo é acionado pela opção “Search Cluster”, uma janela principal a este módulo é exibida como a da Figura 46 (página 118). Esta janela também fornece ao usuários condições de selecionar as ROIs na imagem com uma janela ilustrada na Figura 44(c). Quando as ROIs são selecionadas, a opção *Temperature* da janela principal é habilitada e todas as temperaturas que tiverem as ROIs selecionadas em um único grupo são exibidas para escolha. A opção *Show* na janela principal deste módulo apresenta todas informações deste grupo à temperatura selecionada. As informações do grupo são apresentadas em uma janela com a mesma estrutura da Figura 45(b).

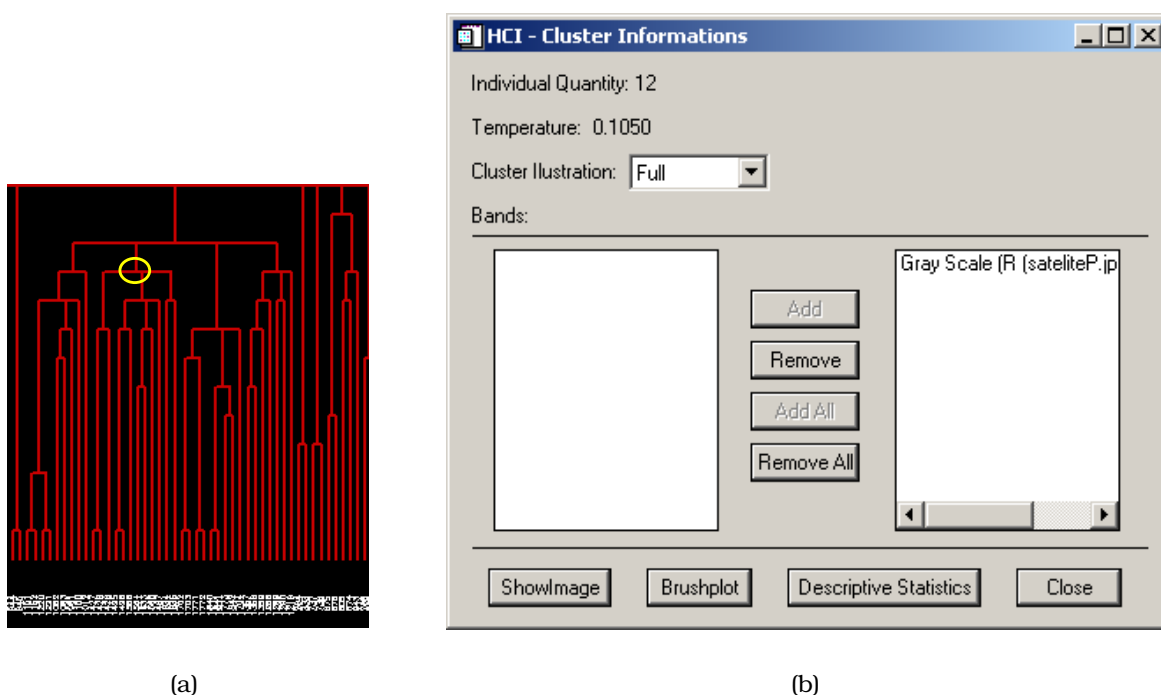


Figura 45: Opção de interação com o dendrograma: (a) área sensível ao clique do mouse e (b) janela de resposta a interação.

Módulo Animação

Este módulo corresponde ao evento “Gerar Animações” (ver Seção 5.3.1). Ele mostra os agrupamento realizados pelo programa SPC através de animações, podendo estas serem salvas em formato com extensão MPEG. A janela principal deste módulo está ilustrada na Figura 47(a) (página 120) e a janela que exibe as animações está ilustrada na Figura 47(b).

Existem duas formas de gerar as animações como apresentada na Tabela 32 (página 101). A primeira exibe uma animação apresentando todos os grupos (segmentos) que compõem a imagem segmentada em uma temperatura selecionada. Esta

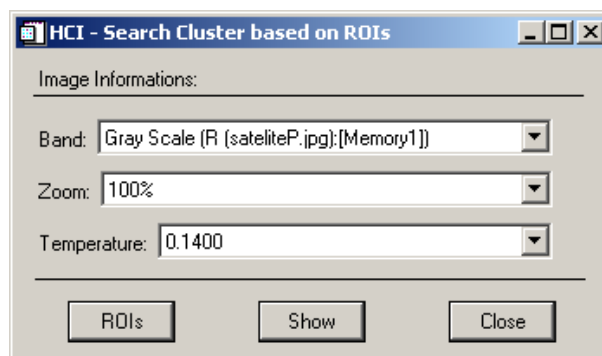


Figura 46: Janela do Módulo Procura Grupo.

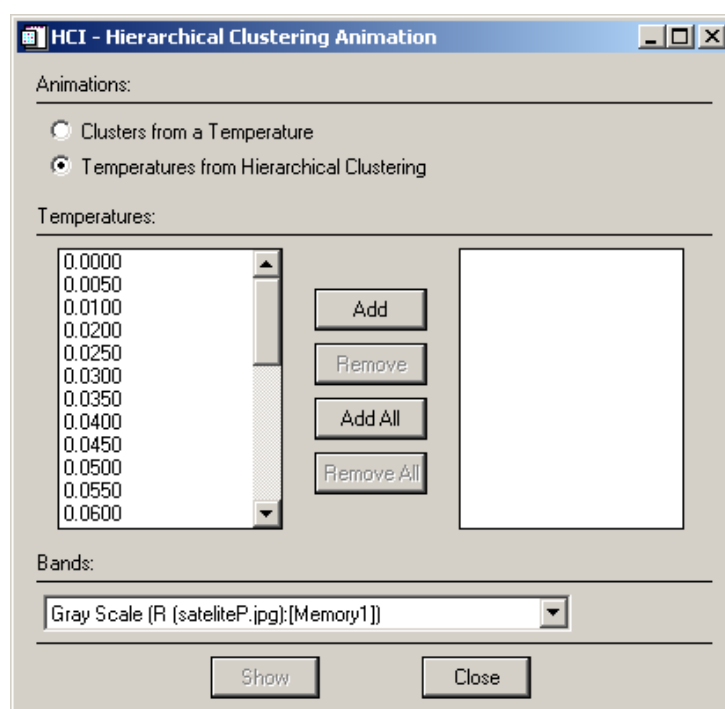
opção está definida na janela principal do módulo como *Cluster from a Temperature*. Para este caso, apenas uma temperatura deve ser selecionada na janela principal. A Figura 47(b) (página 120) ilustra a estrutura da animação para esta opção.

A segunda opção exibe as imagens segmentadas para todas as temperaturas selecionadas na janela principal do módulo. Este tipo de animação fornece uma idéia intuitiva dos grupos formados a cada temperatura. A animação começa pela temperatura mais alta (tende a inúmeros grupos unitários) e continua o processo até a imagem segmentada da temperatura mais baixa (tende a formar um único grupo). Esta segunda opção é realizada quando *Temperatures from Hierarchical Clustering* é selecionado. A Figura 47(c) ilustra a estrutura da animação para esta opção.

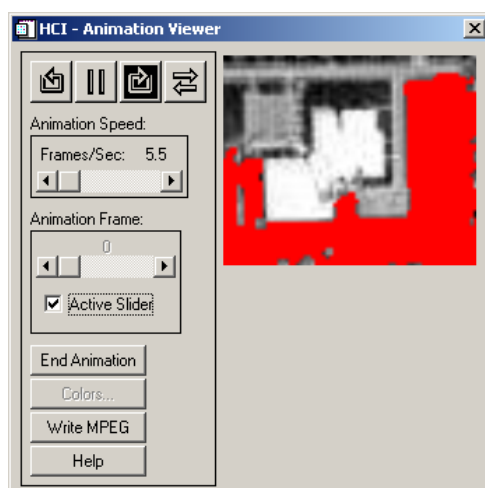
A opção *Band* define qual a banda será utilizada nas animações. A opção *Show* fornece as janelas de animação, só podendo ser habilitada quando alguma temperatura é selecionada.

O sistema desenvolvido é um protótipo para ser analisado em trabalhos futuros. O objetivo principal desta dissertação foi aplicar o agrupamento superparamagnético à segmentação de imagens. Visto dificuldades de manipulação do SPC, foi desenvolvido este sistema. Mas como a técnica realiza um agrupamento hierárquico, pode-se acrescentar novas idéias na exibição dos resultados obtidos. Portanto, nenhum estudo foi realizado sobre a satisfação do usuário, ou seja, as métricas utilizadas para testar a usabilidade do sistema não foram trabalhadas. Esta questão serve de sugestão para trabalhos futuros.

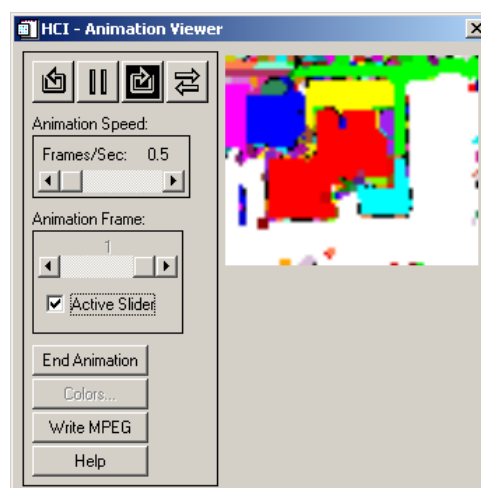
O próximo capítulo apresenta os resultados obtidos diante das pesquisas realizadas na aplicação da segmentação de imagens ao programa SPC.



(a)



(b)



(c)

Figura 47: Janelas do Módulo Animação: (a) principal do módulo, (b) animação por cluster e (c) animação por temperatura.

6 Resultados

Neste capítulo é apresentado o resultado da aplicação do programa SPC otimizado (ver Seção 5.2.3) a uma imagem de sensoriamento remoto. Nele, inicialmente, uma descrição da imagem utilizada é fornecida e, em seguida, os resultados são apresentados.

6.1 Descrição da Imagem

Foi analisada uma imagem do sensor Landsat TM que mostra uma bacia localizada no norte do estado de Wyoming, nos Estados Unidos (ver Figura 48, página 122). A imagem tem 512×512 pixels e possui seis bandas, e faz parte dos dados de demonstração do software ENVI (ver Capítulo 1). O arquivo está no formato de armazenamento nativo do ENVI e se chama `bhtmref.img`.

Como o SPC está limitado a imagens com tamanho pequenos por conta de restrição na manipulação da matriz de dissimilaridade (ver Seção 5.2.1), a imagem foi reduzida para um tamanho de 102×102 pixels. A redução foi feita no ENVI utilizando um fator 0.2 para o número de linhas e de colunas da imagem usando amostragem por vizinho mais próximo. Além disso, algumas áreas da imagem foram selecionadas para uma análise dos dados mais criteriosa.

A Figura 48 exibe uma ilustração das áreas selecionadas na imagem original. A banda 1 foi escolhida para ilustrar todas as imagens da figura, mas o qualidade da imagem no documento foi afetada por causa do formato EPS (utilizado no Latex).

6.2 Resultados

Para todas as imagens analisadas, os parâmetros de entrada necessários ao SPC estão descritos na Tabela 34 (página 122). Nos três exemplos, utilizou-se o SPC otimizado (ver Seção 5.2.3), tendo em uma delas sido utilizado também o SPC com critério de vizinhança KNN. Os arquivos de dados foram gerados definindo a distância Euclidiana e vizinhança-8 na interface. Como as temperaturas utilizadas variam do valor 0.00 até o valor 0.15 com variações 0.005, pode-se ter um total de 31 imagens segmentadas. Porém, a quantidade de imagens segmentadas vai depender dos re-

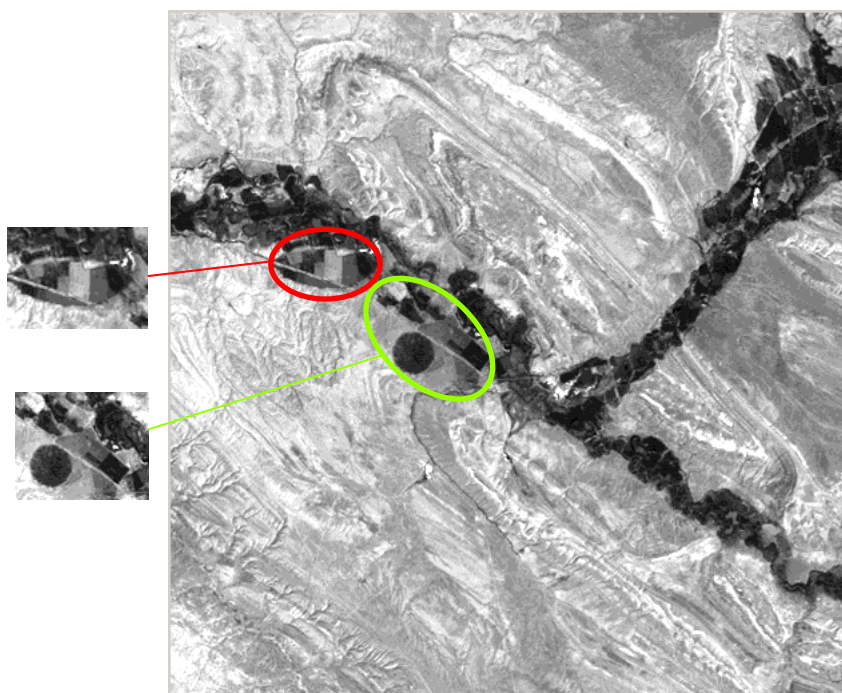


Figura 48: Ilustração geral das imagens de análise.

sultados obtidos a cada temperatura, visto as otimizações realizadas nos arquivos de saída (ver Seção 5.2.2).

O hardware utilizado foi um AMD DURON de 1.3 GHz com memória RAM de 512 MB e HD de 40 GB.

As próximas seções apresentam os resultados para cada imagem analisada.

Dimensions	0
MinTemp	0.00
MaxTemp	0.15
TempStep	0.005
SWCycles	4000
KNearestNeighbours	8
MSTree	
DirectedGrowth	
SaveSuscept	
WriteLabels	
WriteCorFile	~

Tabela 34: Dados de entrada utilizados nas imagens de teste.

6.2.1 Imagem Reduzida

Dentre as 31 temperaturas a serem exibidas na interface, apenas a temperatura 0.015 não foi considerada nos arquivos de saídas. O tempo de processamento do SPC para esta imagem foi de 33 min. Algumas das temperaturas consideradas no arquivos

de saídas são exibidas na Figura 49. Vale salientar que como a imagem foi reduzida para o tamanho 102×102 , a qualidade da segmentação foi afetada.

A Figura 49(a) apresenta a imagem segmentada na temperatura inicial. A visualização desta temperatura foi definida pela opção *full* na interface apresentada na Seção 5.3.2. Como esta temperatura tem valor mínimo, todos os pixels fazem parte de um mesmo grupo (exibido pela cor vermelha em toda a figura).

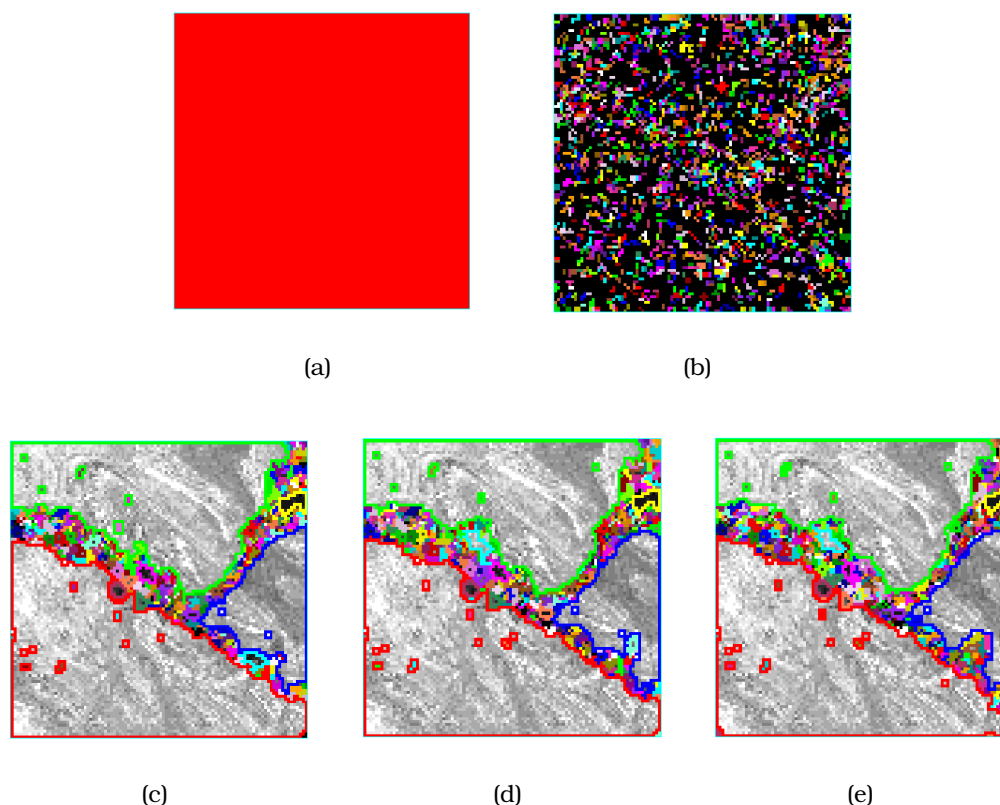


Figura 49: Imagens Segmentadas da Imagem Reduzida por temperaturas: (a) 0.000, (b) 0.150, (c) 0.075, (d) 0.080 e (e) 0.085.

A Figura 49(b) apresenta a imagem segmentada na temperatura final. Ela ilustra a fase de desordem do modelo de Potts. A cor preta, por convenção foi utilizada para representar os clusters unitários. Dado que a quantidade de cores disponíveis pela interface é limitada, a visualização de cada cluster unitário com cores diferentes torna-se um processo inviável dependendo do tamanho da imagem. Esta limitação está apresentada na figura, onde, em algumas regiões separadas espacialmente, as cores se repetem. Isto implica que, por exemplo, dois grupos separados espacialmente são unidos sendo que, na verdade, o que ocorre neste processo é a repetição de cores quando a quantidade já utilizada chega ao limite de cores. Então nestes casos, a imagem segmentada fornece uma ilustração geral, podendo a pesquisa ser mais detalhada a partir de outras opções na interface, como a de procurar grupos ou dendrograma.

As outras figuras apresentam os melhores resultados na segmentação. A Figura 49(c) (página 123) apresenta um resultado interessante, mas na parte inferior direita deixou uma região mais escura ser agrupada com a área mais clara. A Figura 49(d) fornece esta separação em regiões claras e escuras. Quanto mais aumenta a temperatura, mais grupos são formados, ou seja, mais aumenta a quantidade de grupos, podendo este fato ser ilustrado bem na Figura 49(e) para a área analisada.

6.2.2 Primeira Região da Imagem

Esta região da imagem possui o tamanho 100×80 . Por convenção ela será chamada de região 1. Uma ilustração da sua posição na imagem original é mostrada na Figura 48 (página 122), onde a imagem é circundada pela cor verde. Ela foi escolhida por apresentar alguns segmentos bem definidos na imagem original, como o segmento circular (ver Figura 50, página 125).

Os resultados do programa SPC com o critério de vizinhança-8 estão ilustrados na Figura 51 (página 126). A seqüência de figuras ilustra ao caminho de agrupamentos gerados a cada temperatura. Nesta aplicação, das 31 temperaturas a serem exibidas na interface, apenas as temperaturas 0.000, 0.005 e 0.015 não foram consideradas nos arquivos de saídas. A imagem da Figura 51(d) fornece uma boa segmentação. Como a temperatura desta imagem possui seu valor próximo ao das altas temperaturas, a imagem segmentada apresenta uma maior concentração de grupos formados. Um exemplo é o quadrante inferior direito desta imagem.

Na região 1, foi simulada, também, a versão do SPC com o critério KNN (*K-Nearest-Neighbor*) para encontrar os vizinhos de cada pixel. A quantidade máxima de vizinhos estabelecida foi 8. Os resultados de algumas temperaturas estão ilustrados na Figura 52 (página 126). À temperatura 0.040 a imagem segmentada apresenta bons resultados (ver Figura 52(c)). Esta temperatura apresenta seu valor próximo ao das baixas temperaturas, por isto a imagem segmentada apresenta poucos grupos no quadrante inferior direito da imagem.

Uma observação interessante a respeito do SPC com o KNN é o fato dos melhores resultados estarem a baixas temperaturas. Como o algoritmo KNN procura os K vizinhos segundo algum critério de proximidade, os vizinhos de um indivíduo (pixels) tenderão a possuir valores mais baixos (altos) de dissimilaridade (similaridade). Considerando o critério de proximidade utilizado na dissertação, quanto menor for a dissimilaridade entre indivíduos, mais forte é a sua ligação, e mais probabilidade eles têm de se juntarem no passo de formação dos grupos, mesmo a baixas temperaturas. Já na versão do SPC com vizinhança-8, os vizinhos considerados podem ser muito diferentes segundo suas características.

É importante lembrar que versão com o critério KNN só foi aplicada à segmentação de imagens, pois na criação da matriz de dissimilaridade a vizinhança espacial foi

considerada (ver Seção 5.2.1).

Outra informação importante foi a redução brusca no tempo de processamento com o SPC utilizando vizinhança-8. Enquanto a versão com o KNN executou o agrupamento em 131 min, a versão com a vizinhança-8 executou em 23 min, havendo uma redução de 83%.

6.2.3 Segunda Região da Imagem

Esta região da imagem possui 91×66 pixels, e por convenção será chamada de região 2. O tempo de processamento do SPC para esta imagem foi de 19 min. Uma ilustração da sua posição na imagem original é mostrada na Figura 53 (página 126). Ela foi escolhida por apresentar alguns segmentos interessantes para análise na imagem original. Todas as 31 temperaturas foram considerada nos arquivos de saídas para esta imagem.

A Figura 54 (página 127) apresenta a seqüência de temperaturas a formar as segmentações de melhores resultados. Diante das imagens obtidas a da Figura 54(d) apresenta o melhor resultado. Porém, o segmento central apresentado pelo grupo de cor azul não foi segmentado de forma adequada à banda ilustrada. Os segmentos distintos para este grupo só são segmentados à temperatura ilustrada na Figura 54(d). Este fato pode ter ocorrido diante de duas possibilidades:

- As outras bandas favorecerem a união destes dois segmentos, apresentando dissimilaridades mínimas
- Presença de ruídos nas bandas dos segmentos.

O próximo capítulo apresenta as conclusões desta dissertação, acrescentando idéias para trabalhos futuros.



Figura 50: Ilustração da região 1.

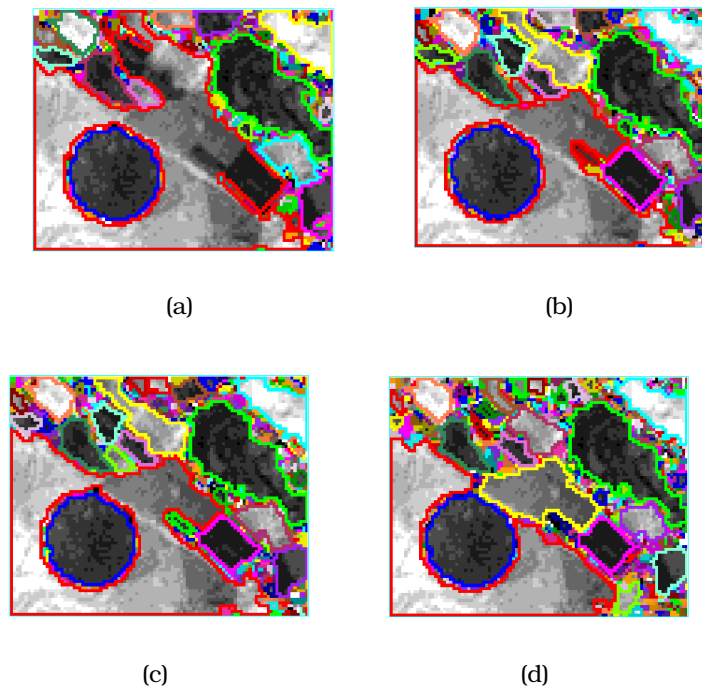


Figura 51: Imagens segmentadas da região 1 pelo critério de vizinhança-8 e com as temperaturas: (a) 0.080, (b) 0.085, (c) 0.090 e (d) 0.100.

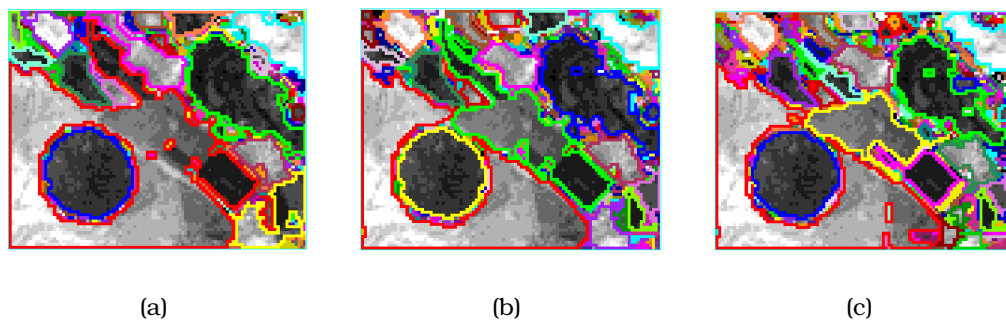


Figura 52: Imagens segmentadas da região 1 pelo critério KNN e com as temperaturas: (a) 0.020, (b) 0.030 e (c) 0.040.



Figura 53: Ilustração da região 2.

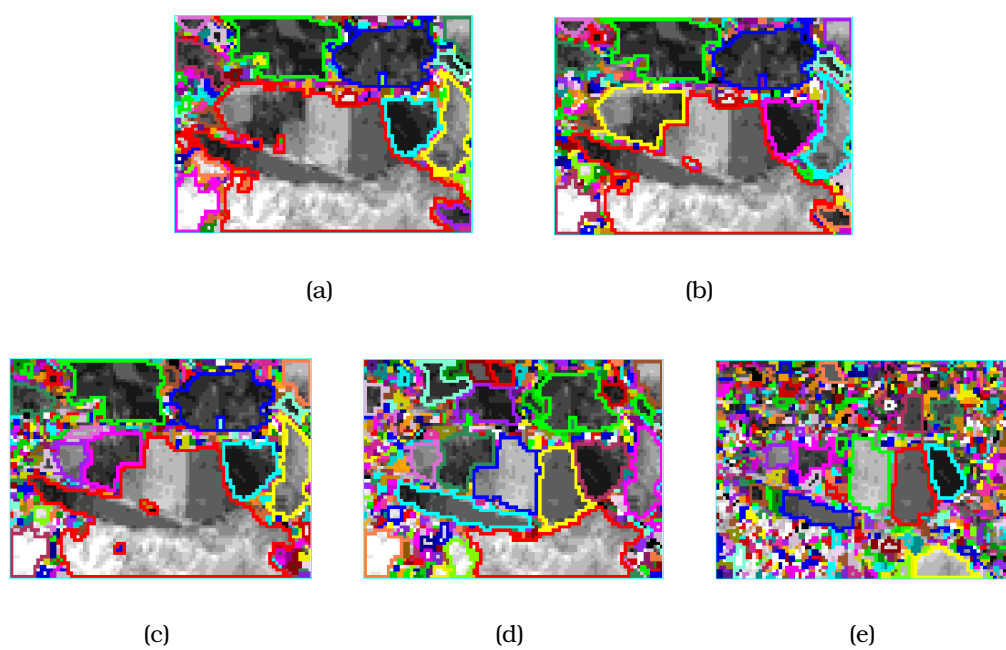


Figura 54: Imagens segmentadas da região 2 por temperatura: (a) 0.090, (b) 0.095, (c) 0.100, (d) 0.105 e (e) 0.125.

7 Conclusões

Este trabalho teve o objetivo de aplicar à segmentação de imagens uma técnica que agrupa dados simulando o comportamento superparamagnético de um modelo físico, o modelo de Potts não homogêneo. Para a localização das configurações típicas deste modelo, a dinâmica de *Swendsen-Wang* foi utilizada. O algoritmo é influenciado por um parâmetro de temperatura. Esta característica resulta num conjunto hierárquico dos dados resultantes, onde a cada temperatura é obtida uma imagem segmentada.

Como os autores do algoritmo desenvolveram a implementação de uma versão acadêmica do mesmo, chamada SPC, este trabalho se dedicou a analisar os resultados e adaptar a versão à segmentação de imagens. Vistas as dificuldades encontradas na manipulação dos arquivos de entrada e saída, necessitando-se de um pré-processamento quanto de um pós-processamento, foi desenvolvido um sistema para amenizar esta etapa.

Para a construção do sistema, otimizações foram realizadas no programa e arquivos de entrada e saída. As otimizações foram necessárias para redução no tempo computacional, dado que as imagens trabalhadas apresentam uma grande quantidade de informação. Como no arquivo de entrada foi estabelecido o critério de vizinhança espacial para construir a matriz de dissimilaridade, a segmentação de imagens também pôde ser realizada com o SPC original para comparações com a versão do programa otimizada.

Foi construída uma interface gráfica para o sistema. Através desta interface inúmeros recursos são disponibilizados, como análise multivariada dos dados provenientes das imagens e exibição dos dados de saída do programa: imagens segmentadas, dendrogramas interativos, animações e busca grupos. Além disso, a interface pode ser utilizada com qualquer algoritmo hierárquico se o mesmo trabalhar com os arquivos de entrada e saída utilizados pela interface.

Na análise dos resultados, utilizou-se imagens de sensoriamento remoto por apresentar volumes de dados grandes e complexos. A imagem utilizada é um dos exemplos presentes no software ENVI. Foram amostradas partes das imagens para uma análise detalhada. As imagens segmentadas apresentam resultados satisfatórios, já que o algoritmo consegue construir grupos visíveis na imagem de forma amigável. As conseqüências das adaptações resultaram numa redução brusca no custo computacional

em relação ao requerido pela versão original.

O sistema desenvolvido é um protótipo e deve ser analisado em trabalhos futuros. Não foram realizados testes para verificar a satisfação do usuário, já que o foco foi descobrir inúmeras formas de se mostrar os resultados obtidos pelo agrupamento hierárquico. Verificar esta satisfação do usuário é de importante valia, servindo como base para a construção de uma versão mais aprimorada do sistema.

Nesta dissertação, analisou-se a aplicação de uma técnica de agrupamento de dados à segmentação de imagens. Como esta técnica é baseada em modelos físicos, as análises de como o comportamento físico influencia nos resultados obtidos foram priorizadas. A qualidade das imagens segmentadas também foi analisada. Visto que um dos propósitos do trabalho foi analisar as segmentações realizadas com base no comportamento físico da técnica, não houve uma comparação com outros algoritmos. Esta comparação é importante para analisar com mais detalhes a qualidade com que a segmentação está sendo realizada, servindo como sugestão a trabalhos futuros.

Os dados trabalhados nesta dissertação só tiveram análises para resultados provenientes das intensidades presentes nas bandas da imagem. Estas análises podem ser estendidas a quaisquer características obtidas. Para utilizá-las com o sistema, elas precisam apresentar valores inteiros e serem atribuídas para cada pixel da imagem.

Como a implementação feita na versão acadêmica restringe a quantidade de dados a serem agrupados, somente imagens relativamente pequenas podem ser utilizadas. Isto ocorre por conta do programa trabalhar com a matriz de dissimilaridade $n \times n$. O tamanho a ser suportado pelo sistema vai depender das configurações presentes no computador. Nas imagens de teste, utilizou-se um computador com 512 MB de RAM. Mas em computadores com 128 MB de RAM, o programa suporta de forma satisfatória imagens com 8000 pixels. Para solucionar este problema, deve-se trabalhar com o conceito de matrizes esparsas.

Referências Bibliográficas


- Ahuja, N. (1982), 'Dot pattern processing using Voronoi neighborhood', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **4**, 336–343. PAMI.
- Banon, G. J. F. (2000), 'Formal introduction to digital image processing'.
URL: <http://iris.sid.inpe.br:1912/rep/dpi.inpe.br/banon/1999/06.21.09.31>
- Blatt, M., S. Wiseman & E. Domany (1996), 'Superparamagnetic clustering of data', *Physical Review Letters* **76**(18), 3251–3254.
- Blatt, M., S. Wiseman & E. Domany (1997), 'Data clustering using a model granular magnet', *Neural Computation* **9**, 1805–1842.
- Bradley, P. S. & U. M. Fayyad (1998), Refining initial points for k-means clustering, *in* 'Proceedings 15th International Conf. on Machine Learning', Morgan Kaufmann, San Francisco, CA, pp. 91–99.
- Bustos, O. H., A. C. Frery & S. Ojeda (1998), 'Strong Markov processes in image modelling', *Brazilian Journal of Probability and Statistics - REBRAPE* **12**(2), 149–194.
- Devroye, L., L. Györfi & G. Lugosi (1996), *A probabilistic theory of pattern recognition*, number 31 *in* 'Applications of Mathematics', Springer-Verlag, New York.
- Domany, E., M. Blatt, Y. Gdalyahu & D. Weinshall (1999), 'Superparamagnetic clustering of data: Application to computer vision', *Computer Physics Communications* **121–122**, 5–12.
- Duda, R. & E. Hart (1973), *Pattern Classification and Scene Analysis*, John Wiley and Sons, New York.
- Duda, R. O., P. E. Hart & D. G. Stork (2001), *Pattern Classification*, 2 edn, John Wiley & Sons, New York.
- ENVI: *Environment for the Visualization of Images* (2003). Última consulta em janeiro de 2004.
URL: <http://www.rsinc.com/envi>
- Everitt, B. S., S. Landau & M. Leese (2001), *Cluster Analysis*, 4 edn, Arnold, New York.
- Facon, J. (1993), *Processamento e Análise de Imagens*, Escola Brasileiro-Argentina de Informática, Embalse, Argentina.
- Ferber, C. & F. Wörgötter (2000), 'Cluster update algorithm and recognition', *Physical Review E* **62**(2), 1461–1464.
- Frery, A. C., E. V. D. Lucca & C. C. Freitas (1999), 'Sar segmentation algorithms: A quantitative assessment', *1999 International Geoscience and Remote Sensing Symposium*.

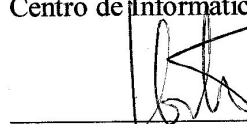
- Frery, A. C., S. J. S. Sant'Anna, N. D. A. Mascarenhas & O. H. Bustos (1997), 'Robust inference techniques for speckle noise reduction in 1-look amplitude SAR images', *Applied Signal Processing* **4**, 61–76.
- Fukunaga, K. (1990), *An Introduction to Statistical Pattern Recognition*, Computer Science and Scientific Computing, second edn, Academic, San Diego.
- Furlan, J. D. (1998), *Modelagem de Objetos através da UML - The Unified Modeling Language*, Makron Books, São Paulo.
- Gdalyahu, Y., D. Weinshall & M. Werman (2001), 'Self-organization in vision: Stochastic clustering for image segmentation, perceptual grouping, and image database organization', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(10), 1053–1074.
- Geman, D. & S. Geman (1984), 'Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**(6), 721–741.
- Gomes, J. & L. Velho (1994), *Computação Gráfica: Imagem*, IMPA/SBM.
- Gonzalez, R. C. & R. E. Woods (2000), *Processamento de Imagens Digitais*, Edgard Blücher, São Paulo.
- Gordon, A. D. (1999), *Classification*, 2 edn, Chapman & Hall, London.
- Gower, J. C. (1985), *Measures of similarity, dissimilarity and distance*, Encyclopaedia of Statistical Sciences, Volume 5, Wiley, New York.
- Gower, J. C. & P. Legendre (1986), 'Metric and euclidean properties of dissimilarity coefficients', *Journal of Classification* **5**, 5–48.
- Harada, Y. K., M. Miyakoshi & M. Shimbo (1994), 'A fuzzy clustering method for automatic segmentation of color scenes', *Japanese Journal of Fuzzy Theory and Systems* **6**(5), 651–672.
- Haralick, R. M. & L. G. Shapiro (1985), 'Image segmentation techniques', *Computer Vision, Graphics and Image Processing* **29**(1), 100–132.
- IDL: Interactive Data Language* (2003). Última consulta em janeiro de 2004.
URL: <http://www.rsinc.com/idl>
- Jain, A. K. (1989), *Fundamentals of Digital Image Processing*, Prentice-Hall International Editions, Englewood Cliffs, NJ.
- Johnson, R. A. & D. W. Wichern (1992), *Applied Multivariate Statistical Analysis*, 3 edn, Prentice Hall, New Jersey.
- Kaufman, L. & P. J. Rousseeuw (1990), *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley & Sons, New York.
- Kurita, T. (1991), 'An efficient agglomerative clustering algorithm using a heap', *Pattern Recognition* **24**(3), 205–209.
- Lance, G. N. & W. T. Willians (1966), 'Computer programs for hierarchical polythetic classification', *Computer Journal* **9**, 60–64.

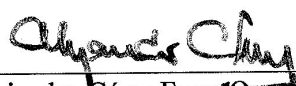
- Larson, R. C. & G. Sadiq (1983), 'Facility locations with the Mahanttan metric in the presence of barriers to travel', *Operation Research* **31**, 652–699.
- Lucca, E. V. D. (1998), Avaliação e comparação de algoritmos de segmentação de imagens de radar de abertura sintética, Dissertação de mestrado em sensoriamento remoto, Instituto Nacional de Pesquisas, São José dos Campos, SP.
- Lucchese, L. & S. K. Mitra (2001), 'Color image segmentation: A state-of-the-art survey', *Proc. of the Indian National Science Academy (INSA-A)* **67**(2), 207–221.
- Mascarenhas, N. D. A. & F. R. D. Velasco (1989), *Processamento Digital de Imagens*, Kapelusz, Buenos Aires, Argentina.
- Mascarenhas, N. D. A., L. V. Dutra & A. C. Frery (2000), The effect of quantization on SAR parameter estimation, in 'International Geoscience and Remote Sensing Symposium: The Role of Remote Sensing in Managing the Global Environment', IEEE, IEEE Press, Hawaii. CD-ROM.
- McQueen, J. (1967), 'Some methods for classification and analysis of multivariate observations', *5-th Berkeley Symposium on mathematics* **1**, S. 281–298.
- Metropolis, N., A. W. Rosebluth, M. N. Rosebluth, A. H. Teller & E. Teller (1953), 'Equations of state calculations by fast computing machines', *Journal of Chemical Physics* pp. 1087–1092.
- Metropolis, N. & S. Ulam (1949), 'The Monte Carlo method', *Journal of the American Statistical Association* pp. 335–341.
- Morris, S. A., B. Asnake & G. G. Yen (2003), 'Dendrogram seriation using simulated annealing', *Information Visualization* **2**(2), 95–104.
- Neirotti, J. P., S. M. Kurcbart & N. Caticha (2003), 'Superparamagnetic segmentation by excitable neural systems', *Physical Review E* **68**(31911), 1–9.
- Neubert, M. & G. Meinel (2003), 'Evaluation of segmentation programs for high resolution remote sensing applications', *Proceedings of the ISPRS Joint Workshop High Resolution Mapping from Space*.
- Oliveira, M. G. S. (2004), Sistema para agrupamento de dados baseado no comportamento superparamagnético do modelo de potts, Dissertação de mestrado em ciência da computação, Centro de Informática, Universidade Federal de Pernambuco, Recife, PE.
- Peña, J. M., J. A. Lozano & P. Larrañaga (1999), 'An empirical comparison of four initialization methods for the k-means algorithm', *Pattern Recognition Letters* **20**, 1027–1040.
- Peng, S., B. Urbanc, L. Cruz, B. T. Hyman & H. E. Stanley (2003), 'Neuron recognition by parallel potts segmentation', *PNAS* **100**(7), 3847–3852.
- Pickard, D. K. (1987), 'Inference for discrete Markov fields: The simplest nontrivial case', *Journal of the American Statistical Association* **82**(1), 90–96.
- Pompilho, S. (2002), *Análise Essencial - Guia prático de Análise de Sistemas*, Editora Ciência Moderna, Rio de Janeiro.

- Ripley, B. D. (1996), *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge.
- Sharma, M. (2001), Performance evaluation of image segmentation and texture extraction methods in scene analysis, Master of philosophy in computer science, University of Exeter, Exeter, UK.
- Swendsen, R. (1991), 'Acceleration methods for Monte Carlo computer simulations', *Computer Physics Communications* **65**(1/3), 281–288.
- Swendsen, R. & J. Wang (1987), 'Nonuniversal critical dynamics in Monte Carlo simulations', *Physical Review Letters* **58**(2), 86–88.
- Wiseman, S., M. Blatt & E. Domany (1998), 'Superparamagnetic clustering of data', *Physical Review E* **57**(4), 3767–3783.
- Wu, F. Y. (1982), 'The Potts model', *Reviews of Modern Physics* **54**(1), 235–268.
- Yourdan, E. (1990), *Análise Estruturada Moderna*, Editora Campus, Rio de Janeiro.

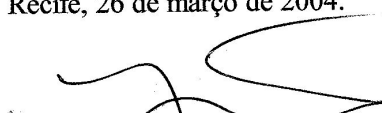
Dissertação de Mestrado apresentada por **Michelle Matos Horta** à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título , “**Sistema para Segmentação de Imagens por Agrupamento Hierárquico Baseado no Comportamento Superparamagnético do Modelo de Potts**”, orientada pelo Prof. **Alejandro César Frery Orgambide** e aprovada pela Banca Examinadora formada pelos professores:


Prof. Fernando da Fonseca de Souza
Centro de Informática / UEPE


Prof. Luciano da Fontoura Costa
Departamento de Física e Informática / USP


Prof. Alejandro César Frery Orgambide
Departamento de Tecnologia da Informação / UFAL

Visto e permitida a impressão.
Recife, 26 de março de 2004.


Prof. JAELSON FREIRE BRELAZ DE CASTRO
Coordenador da Pós-Graduação em Ciência da Computação do
Centro de Informática da Universidade Federal de Pernambuco.