Adequando consultas por similaridade para reduzir a descontinuidade semântica na recuperação de imagens por conteúdo

Humberto Luiz Razente

Adequando consultas por similaridade para reduzir a descontinuidade semântica na recuperação de imagens por conteúdo

Humberto Luiz Razente

Orientador: ***Prof. Dr. Caetano Traina Júnior***

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências - Ciências de Computação e Matemática Computacional.

USP – São Carlos
Julho de 2006

Trabalho realizado com auxílio financeiro da Fundação de Amparo à Pesquisa do Estado de São Paulo – FAPESP (bolsa de doutorado, processo nº 2006/00336-5, de janeiro/2007 a outubro/2009), da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES (bolsa de doutorado institucional, de março/2006 a dezembro/2006) e do Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq.

À minha querida esposa Camila.

Aos meus pais, Pedro e Vilma,
e meus irmãos, Edson e Júlio.

Ao meu avô, Antônio. Em memória dos
meus avós Júlio, Leonora e Augusta.

Agradecimentos

A Deus por estar sempre comigo. Aos meus pais, por terem me incentivado no caminho das ciências e por sempre terem feito tudo o que estava ao alcance para que eu pudesse chegar até esta tese. A minha querida esposa Camila, por estar sempre ao meu lado, pela sua dedicação e apoio durante todos esses anos, e pela grande contribuição nos vários trabalhos desenvolvidos durante a realização desta tese. Ao meu orientador, professor Caetano, pela orientação durante o mestrado e o doutorado, agradeço seu apoio, incentivo e confiança, seus conselhos foram essenciais para a minha formação como pesquisador. A professora Agma, pela co-orientação e pelos conselhos preciosos. A professora Franklina M. B. Toledo pelo auxílio com os métodos de programação linear e não-linear e com os métodos de avaliação de problemas de otimização. Aos professores Josiane M. Bueno (*in memorian*), Eduardo R. Hruschka e Cristina D. A. Ciferri pela orientação no Programa de Aperfeiçoamento de Ensino. A professora Cláudia A. Martins (UFMT), pela amizade, incentivo e pelas valiosas cartas de recomendação. Aos professores Mauro Biajiz e Ricardo Ciferri (UFSCAR) pelas dicas nas reuniões do grupo de pesquisa. Ao professor Christos Faloutsos (Carnegie Mellon) pelas sugestões. Ao professor Vassilis Tsotras (University of California em Riverside) por me receber em seu grupo de pesquisa e me orientar durante meu estágio sanduíche. Ao amigo Marcos R. Vieira pela grande ajuda durante minha estadia em Riverside. Aos amigos que me receberam em seus lares (mais conhecidos como repúblicas), Caio e Guilherme, Pedro, Marcelo Castoldi/Carlisson/Leandro/Moussa, Daniel/Isa, Marcos/Marios, Georgos/Ted. Aos amigos que fiz em Riverside, Petko, Marios, Georgos, Ted, Rubens, Ricardo e Thiago. Aos companheiros de laboratório da "velha guarda", Enzo e Thatyana, Adriano, Josiel, Marcos, Fábio, Renato, Elisangela, Ana Paula, Elaine, Roberto, e aos companheiros da "nova guarda", Caio, Daniel, Gabriel, Robson, Pedro, Carol, Marcela, Paterlini, Rodrigo, João Paulo, Ives, Mônica, Luciana, André, Júnior, Sérgio, Willian, Pedro, Bruno. Aos funcionários e professores do ICMC, em especial à secretaria de pós-graduação, assistência acadêmica, secretaria do departamento de computação, seção de eventos, seção de pessoal e setor financeiro. Ao CCMC-ICMC-USP, CNPq, CAPES e FAPESP pelos auxílios financeiros.

Resumo

Com o crescente aumento no número de imagens geradas em mídias digitais surgiu a necessidade do desenvolvimento de novas técnicas de recuperação desses dados. Um critério de busca que pode ser utilizado na recuperação das imagens é o da dissimilaridade, no qual o usuário deseja recuperar as imagens semelhantes à uma imagem de consulta. Para a realização das consultas são empregados vetores de características extraídos das imagens e funções de distância para medir a dissimilaridade entre pares desses vetores. Infelizmente, a busca por conteúdo de imagens em consultas simples tende a gerar resultados que não correspondem ao interesse do usuário misturados aos resultados significativos encontrados, pois em geral há uma descontinuidade semântica entre as características extraídas automaticamente e a subjetividade da interpretação humana. Com o intuito de tratar esse problema, diversos métodos foram propostos para a diminuição da descontinuidade semântica. O foco principal desta tese é o desenvolvimento de métodos escaláveis para a redução da descontinuidade semântica em sistemas recuperação de imagens por conteúdo em tempo real. Nesta sentido, são apresentados: a formalização de consultas por similaridade que permitem a utilização de múltiplos centros de consulta em espaços métricos como base para métodos de realimentação de relevância; um método exato para otimização dessas consultas nesses espaços; e um modelo para tratamento da diversidade em consultas por similaridade e heurísticas para sua otimização.

Palavras-chave: recuperação de imagens por conteúdo, descontinuidade semântica, consultas por similaridade agregada, diversidade em consultas aos vizinhos mais próximos.

Razente, Humberto Luiz. *Adequando Consultas por Similaridade para Reduzir a Descontinuidade Semântica na Recuperação de Imagens por Conteúdo* (2009). Tese de doutorado, Instituto de Ciências Matemáticas e de Computação – ICMC, Universidade de São Paulo – USP, 125 páginas.

Abstract

The increasing number of images captured in digital media fostered the developmet of new methods for the recovery of these images. Dissimilarity is a criteria that can be used for image retrieval, where the results are images that are similar to a given reference. The queries are based on feature vectors automatically extracted from the images and on distance functions to measure the dissimilarity between pair of vectors. Unfortunately, the search for images in simple queries may result in images that do not fulfill the user interest together with meaningful images, due to the semantic gap between the image features and to the subjectivity of the human interpretation. This problem leaded to the development of many methods to deal with the semantic gap. The focus of this thesis is the development of scalable methods aiming the semantic gap reduction in real time for content-based image retrieval systems. For this purpose, we present the formal definition of similarity queries based on multiple query centers in metric spaces to be used in relevance feedback methods, an exact method to optimize these queries and a model to deal with diversity in nearest neighbor queries including heuristics for its optimization.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

Lista de Siglas

ALOI	*Amsterdam Library of Object Images*
ANNq	*aggregate nearest neighbor query*
ARq	*aggregate range query*
CBIR	*content-based image retrieval*
EV	expansão da vizinhança
GRASP	*greedy randomized adaptive search procedure*
MAM	método de acesso métrico
MASS	*metric aggregate similarity search*
MBM	*minimum bounding method*
MBR	*minimum bounding rectangle*
MDP	*maximum diversity problem*
NDNq	*nearest diverse neighbor query*
NNq	*nearest neighbor query*
PACS	*picture archiving and communication system*
PR	*path relinking*
QEX	*query expansion*
QPM	*query point movement*
RF	*relevance feedback*
ROC	*receiver operating characteristic*
Rq	*range query*
SGBD	sistema de gerenciamento de banco de dados
SVM	*support vector machine*
TBIR	*text-based image retrieval*
TF	*term frequency*

Capítulo

1

Introdução

Com a crescente evolução dos diversos dispositivos de aquisição de imagens em meios digitais, tanto para uso pessoal (como as câmeras fotográficas digitais), quanto para equipamentos de uso profissional (como aparelhos de raios-x digitais e tomógrafos em ambientes hospitalares), surgiu a necessidade do desenvolvimento de técnicas de recuperação capazes de lidar de maneira eficiente com o grande volume de dados gerado. Por exemplo, o armazenamento de imagens de exames em um grande hospital pode ser da ordem de dezenas ou centenas de *terabytes* a cada ano. O simples armazenamento dessa grande quantidade de imagens, sem a criação de mecanismos eficientes de recuperação, pode levá-las a possivelmente nunca mais serem acessadas [Fayyad e Uthurusamy, 2002].

Há duas abordagens clássicas para a recuperação de imagens: a abordagem baseada em textos e a abordagem baseada em conteúdo [Han e Kamber, 2006]. A abordagem baseada em textos (*Text-Based Image Retrieval* – TBIR) aplica técnicas de recuperação de textos às anotações, descrições feitas para cada imagem, legendas, palavras-chaves ou meta-dados como tipo de exame, corte, número de pixels, bits por pixel, equipamento de geração, data e hora da criação, posicionamento global, caminho no sistema de arquivos, configurações de brilho e contraste, entre outros. Essa abordagem teve seu desenvolvimento iniciado na década de 1980 [Chang e Fu, 1980, Chang e Kunii, 1981] e vem sendo pesquisada desde então [Zhang et al., 2005]. Entretanto, ela apresenta um alto custo associado à exigência de que todas as imagens devem ser analisadas, interpretadas e descritas por um analista humano, sendo que a existência de uma grande quantidade de imagens dificulta a diferenciação de cada uma em suas descrições e a representação real dos detalhes que diferenciam cada uma. Além disso, diferentes interpretações podem ser dadas

acerca do conteúdo de uma mesma imagem devido à subjetividade da percepção humana [Müller et al., 2004]. Com a explosão da quantidade de imagens capturadas, tornou-se necessário o uso de técnicas automáticas que permitam a indexação e recuperação de imagens, independentemente dos interesses dos usuários que efetuam as descrições e do grau de detalhe das descrições efetuadas. Nessas técnicas, em geral, a computação de descrições automáticas resulta em baixa qualidade discriminativa.

O agravamento desses problemas, causado pela rápida expansão do volume e abrangência das informações visuais, gerou a necessidade do desenvolvimento de novas técnicas que pudessem gerenciar de um modo mais eficiente e preciso a representação e a recuperação de imagens. Uma abordagem que tem sido considerada por diversos pesquisadores é denominada Recuperação de Imagens Baseada em Conteúdo (*Content-Based Image Retrieval* – CBIR), na qual descritores obtidos de maneira automática, tais como histogramas de cores, textura, topologia, formas dos objetos e suas disposições nas imagens são utilizadas na comparação das mesmas. Um trabalho pioneiro dessa abordagem foi publicado em [Chang e Liu, 1984] e desde então diversas propostas foram estudadas [Müller et al., 2004, Liu et al., 2007]. Dado que o usuário possui uma imagem, a busca por uma cópia idêntica em um conjunto de imagens é uma consulta de pouca utilidade. Nessa abordagem, o critério utilizado para recuperação das imagens é o da dissimilaridade, no qual são procuradas as imagens menos dissimilares a uma imagem de referência. Essas consultas são denominadas consultas por similaridade.

As técnicas baseadas em busca por conteúdo dependem de algoritmos de processamento de imagens para extrair automaticamente as características mais relevantes de cada imagem. Cada característica é usualmente um valor ou conjunto de valores numéricos, e o conjunto de características extraído de uma imagem é chamado de um vetor de características. Exemplos comuns de características extraídas de imagens são os descritores de cor, textura e forma. Os vetores de características são utilizados para as operações de computação da dissimilaridade entre as imagens, ao invés das imagens propriamente ditas, sendo que a avaliação da dissimilaridade é dada por uma função de distância. Um exemplo dessas funções é a função de distância Euclidiana.

Há duas abordagens principais para a manipulação de imagens por conteúdo de acordo com o domínio de dados dos vetores de características:

- Modelo de espaço multidimensional: os elementos de dados são representados por vetores numéricos de tamanho fixo e são tratados como pontos em um espaço multidimensional. Nesse modelo, a dissimilaridade pode ser computada por funções de distância como as da família Minkowski (L_p), e a distribuição dos valores nas dimensões pode ser usada para criar hierarquias que permitem a otimização de diversos tipos de consulta.

- Modelo de espaço métrico: abrange dados cuja noção de dissimilaridade pode ser mais complexa que vetores de dimensão fixa. É representado por um domínio de dados e uma função de distância, nos quais pode não haver informações sobre dimensões e ordem. Nesse caso, a dissimilaridade é altamente dependente do domínio dos dados, e deve ser computada por uma função de distância. Os vetores numéricos de tamanho fixo e as funções de distância que lidam com esses vetores também podem ser considerados nesse modelo.

O problema de busca é geralmente limitado pelo domínio dos dados, pelo método de comparação de elementos e pela especificação das consultas. O tratamento dos dados pelo modelo de espaço métrico apresenta vantagens na generalidade, uma vez que muitos domínios e estratégias de busca permitem a sua utilização. Além disso, a fundamentação matemática do modelo permite a otimização de diversos problemas.

A otimização de consultas é geralmente dada pela indexação dos dados em métodos de acesso especializados. Considerando o modelo de espaço multidimensional, há diversos métodos que permitem uma vasta gama de consultas complexas, entre elas as consultas topológicas, direcionais, por similaridade, entre outras. Entretanto, esses métodos têm seu desempenho degradado rapidamente com o aumento do número de dimensões, diminuindo sua eficácia em características extraídas de imagens que, em geral, são de alta dimensionalidade. Por outro lado, os métodos de acesso que consideram o modelo de espaço métrico são mais robustos para a realização de consultas baseadas na dissimilaridade.

1.1 Definição do Problema

Embora muitas pesquisas tenham sido realizadas para o desenvolvimento de técnicas de recuperação de imagens baseada em conteúdo, a maioria delas ainda é incapaz de extrair todo o conteúdo de uma imagem, e muitas vezes falha na tarefa de representar adequadamente o seu significado. O principal problema está relacionado à descontinuidade semântica entre as características de baixo nível, extraídas automaticamente de imagens, e a subjetividade da interpretação humana. Com o intuito de tratar esse problema, tem sido considerada a utilização de técnicas de realimentação de relevância (*Relevance Feedback*) para recuperação de imagens por conteúdo. Nessas técnicas, exemplos considerados positivos e negativos são informados ao sistema pelo usuário logo após a realização de uma consulta, com o intuito de permitir a derivação de uma representação da intenção do usuário que possa melhorar as respostas de consultas futuras.

Durante mais de uma década, dezenas de trabalhos trataram de técnicas de realimentação de relevância para recuperação de imagens por conteúdo [Zhou e Huang, 2003, Liu et al., 2007, Heesch, 2008]. Vários desses trabalhos apresentam abordagens para me-

lhorar a semântica das consultas por conteúdo, passando pelo treinamento de classificadores e calibração de funções de distância, métodos que fazem seleção de atributos, modelos que permitem realimentação de relevância, entre outros. Entretanto, muitos desses métodos não são escaláveis para grandes conjuntos de imagens.

1.2 Objetivos

O foco principal desta tese é o estudo e o desenvolvimento de métodos otimizados para a redução da descontinuidade semântica em sistemas de recuperação de imagens por conteúdo. Para tanto, esta tese teve por objetivo o desenvolvimento de um método de realimentação de relevância eficiente e eficaz e o estudo do relacionamento entre os elementos que compõem o resultado das consultas típicas. Os métodos de realimentação de relevância têm por objetivo melhorar a semântica das consultas em tempo real, podendo então ser utilizados em sistemas de recuperação de imagens. A hipótese foi que esses métodos são passíveis de otimização se considerados como imersos em espaços métricos. O desenvolvimento de métodos otimizados é um passo fundamental para uma futura integração com os sistemas de gerenciamento de banco de dados.

As consultas por similaridade típicas baseiam-se em apenas um centro de consulta, e essas consultas já foram extensivamente exploradas [Zezula et al., 2006, Samet, 2006]. Por outro lado, a realimentação de relevância pode resultar em consultas com múltiplos centros. Assim, nesta tese, definiu-se como tratar essas consultas no modelo métrico, bem como suas propriedades e otimização. Há vários modos de realizar consultas com mais de um centro, e o objetivo foi a definição de um modelo que permitisse a sua otimização. Em geral, o modo como os elementos de um conjunto resposta se relacionam também afeta a semântica, uma vez que a diversidade entre os elementos da resposta tem impacto na qualidade dos resultados. A hipótese foi que a avaliação do relacionamento entre elementos próximos a um elemento de consulta, permitindo a seleção de elementos diversos, tem potencial para auxiliar a reduzir a descontinuidade semântica. Entretanto, a avaliação de diversidade em consultas em bases de imagens é um problema computacionalmente complexo, principalmente em consultas por similaridade.

Assim, nesta tese foram explorados os seguintes problemas:

- o desenvolvimento de um modelo de realimentação de relevância em espaços métricos, baseado em consultas de múltiplos centros;
- a otimização de consultas de múltiplos centros em métodos de acesso para espaços métricos, permitindo a realização dessas consultas em sistemas de gerenciamento de banco de dados;

- o tratamento da diversidade em consultas por similaridade como modo de melhorar a qualidade semântica dos sistemas de recuperação de imagens por conteúdo.

1.3 Contribuições

Esta tese trata da definição, otimização e melhoria da qualidade semântica de métodos de realimentação de relevância baseados em consultas de múltiplos centros para conjuntos de dados imersos em espaços métricos, tornando esses métodos escaláveis para grandes conjuntos de dados. Este trabalho apresenta três contribuições principais:

- a definição das consultas por similaridade agregada como uma generalização das consultas por similaridade baseadas em um único centro e suas propriedades, bem como sua aplicação em métodos de realimentação de relevância em consultas por conteúdo de imagens;
- um método exato para otimização de consultas por similaridade agregada em métodos de acesso para espaços métricos, desenvolvido com base nas propriedades desses espaços;
- um modelo para tratamento da diversidade em consultas por similaridade e algoritmos para sua otimização.

Para comprovar as hipóteses levantadas, foram realizados diversos experimentos, que são apresentados juntamente com os métodos desenvolvidos. A avaliação empírica desses métodos é dada pela comparação com os métodos atuais descritos na literatura. Todos os métodos desenvolvidos podem ser empregados em métodos de acesso para espaços métricos já existentes, como os métodos M-tree e Slim-tree.

1.4 Organização

Esta tese está organizada em sete capítulos conforme segue:

- Capítulo 1. Introdução, definição do problema, objetivos e contribuições desta tese;
- Capítulo 2. Apresenta os principais temas relacionados à recuperação de imagens baseada em conteúdo, entre eles a extração de características, medidas de dissimilaridade, principais consultas e métodos de acesso;

- Capítulo 3. Apresenta uma revisão bibliográfica sobre as abordagens para o tratamento da descontinuidade semântica em buscas por conteúdo de imagens, com maior atenção às técnicas de realimentação de relevância;

- Capítulo 4. Introduz as consultas por similaridade agregada e suas propriedades, descreve o método para otimização das consultas por similaridade agregada, e apresenta experimentos que comprovam sua eficácia em métodos de realimentação de relevância e eficiência em relação aos métodos existentes na literatura;

- Capítulo 5. Apresenta o modelo para tratamento da diversidade em consultas por similaridade limitadas por número de elementos em espaços métricos, bem como as heurísticas desenvolvidas para sua execução;

- Capítulo 6. Considerações finais e propostas para pesquisas futuras.

Capítulo

2

Recuperação de Imagens Baseada em Conteúdo

A recuperação de imagens tem sido uma área de pesquisa muito ativa em banco de dados e visão computacional na última década. A disponibilidade de grandes e crescentes quantidades de imagens geradas e armazenadas exige o desenvolvimento de formas de acesso que vão além de consultas baseadas em textos ou chaves. Muitos sistemas foram propostos para tratar do armazenamento e da recuperação de imagens de forma eficiente e eficaz, porém uma solução ótima ainda está longe de ser alcançada. Em se tratando do armazenamento e recuperação de imagens de exames médicos, a maioria dos sistemas de arquivamento e comunicação de imagens (*picture archiving and communication systems – PACS*) utiliza apenas dados textuais ou numéricos para acessar imagens de exames de pacientes, os quais são inseridos manualmente nos sistemas.

Ao considerar o diagnóstico por imagens, pode-se usar um sistema PACS para a consulta de imagens baseada em dados textuais. As anotações textuais em uma imagem dependem de um processo manual e demorado que podem apresentar variações de um usuário para outro, dependendo do contexto em que cada usuário utiliza a imagem. Entretanto, a possibilidade de recuperar imagens por similaridade dos seus conteúdos é uma característica desejável. A recuperação de imagens baseada em conteúdo (*content-based image retrieval – CBIR*) baseia-se principalmente em características extraídas automaticamente de cada imagem.

Grande parte dos sistemas de recuperação de imagens baseada em conteúdo tem arquitetura similar, composta por métodos de interação com o usuário, armazenamento, indexação e extração de características, nos quais um mecanismo de recuperação de ima-

gens integra todos os componentes necessários para a realização de consultas por conteúdo de imagens [Müller et al., 2004], como apresentado na Figura 2.1.

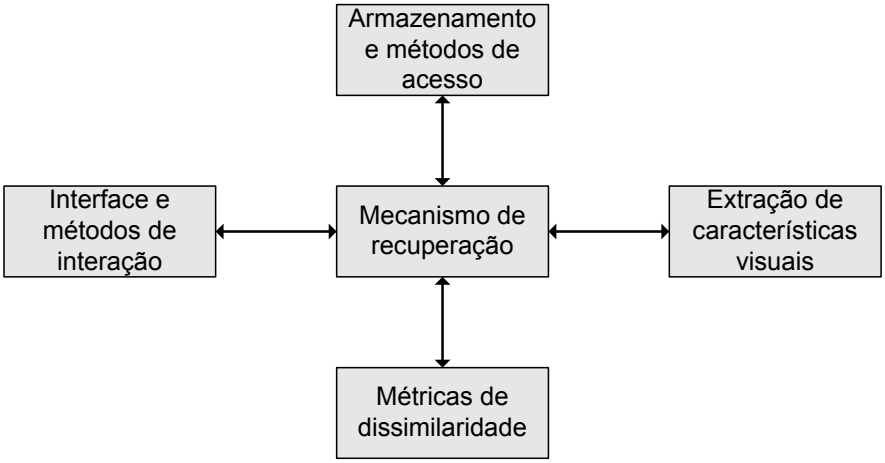

Figura 2.1: Principais componentes dos sistemas de recuperação de imagens baseada em conteúdo [Müller et al., 2004].

O processo geral de recuperação de imagens por conteúdo é apresentado na Figura 2.2, com destaque para dois fluxos de dados: um para a fase de processamento da inclusão de novas imagens e outro para a fase de consultas por similaridade. As imagens são geradas no processo de aquisição nos diversos equipamentos e ambientes disponíveis, e em seguida elas são armazenadas na base de imagens e suas características são extraídas e indexadas na base de características. Para a realização de uma consulta por conteúdo, a imagem de referência deve passar pelo mesmo processo de extração de características, permitindo o cálculo de similaridade. Finalmente, o resultado da consulta é apresentado e uma realimentação acerca da relevância de cada imagem retornada pode ser fornecida visando o refinamento de consultas futuras [Smeulders et al., 2000].

A seguir são apresentados os principais conceitos envolvidos na recuperação de imagens baseada em conteúdo. Uma visão geral dos componentes dos sistemas de recuperação de imagens baseada em conteúdo pode ser encontrada nos seguintes trabalhos de revisão da área [Long et al., 2003, Müller et al., 2004, Datta et al., 2008].

Este capítulo está estruturado da seguinte forma. A Seção 2.1 trata dos métodos de extração de características de imagens e a Seção 2.2 apresenta as principais medidas de similaridade utilizadas nesses domínios. A Seção 2.3 aborda brevemente as principais consultas por similaridade e a Seção 2.4 apresenta como é possível otimizar essas consultas. Finalmente, a Seção 2.5 apresenta as considerações finais do capítulo.

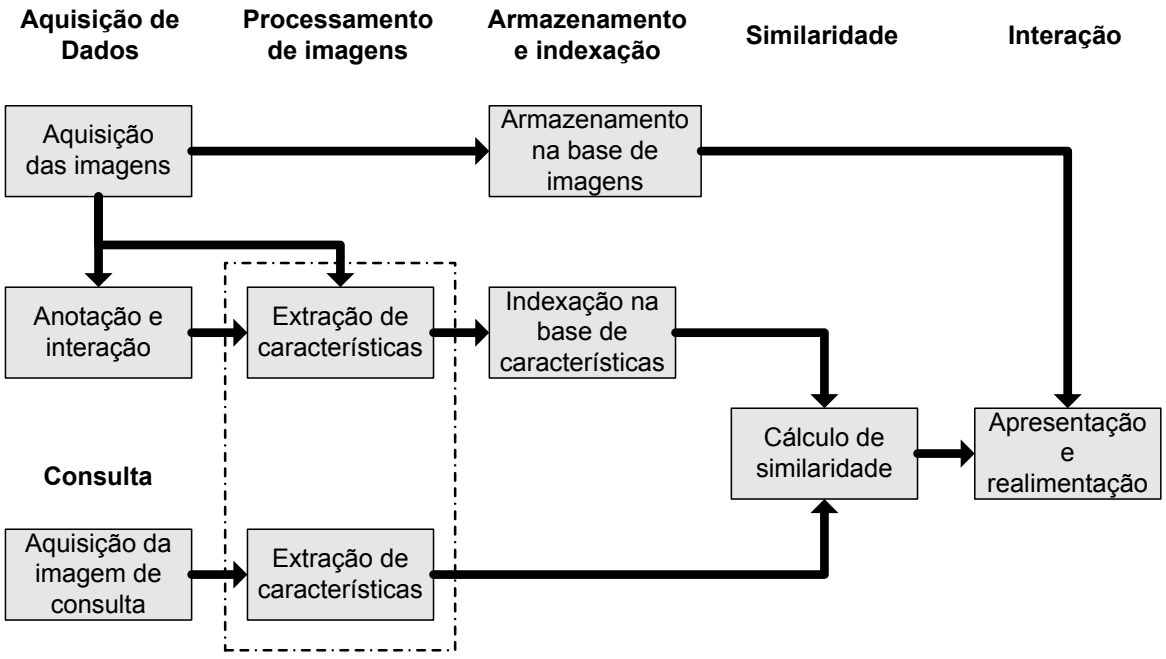

Figura 2.2: Ilustração do fluxo de dados entre os módulos de um sistema de recuperação de imagens por conteúdo [Smeulders et al., 2000].

2.1 Extração de Características

Para a realização de buscas por conteúdo, algoritmos de extração de características devem ser empregados para extrair automaticamente vetores de características das imagens, isto é, sem intervenção humana. Os vetores de características são utilizados na recuperação por conteúdo, ao invés das imagens propriamente ditas. Eakins e Graham em [Eakins e Graham, 1999] definem três tipos de consultas por conteúdo de imagens:

- Tipo 1: recuperação por meio da extração de características primitivas como cor, textura, forma ou a localização espacial de elementos da imagem. Em geral é utilizada em consultas baseadas em exemplo, como: "encontre imagens semelhantes a uma dada imagem";
- Tipo 2: recuperação de imagens de um dado tipo que é identificado por características derivadas com algum grau de inferência lógica, por exemplo, "encontre imagens de bicicletas";
- Tipo 3: recuperação por meio de atributos abstratos que envolvem uma quantidade significativa de raciocínio sobre a intenção do usuário, por exemplo, "encontre imagens de pessoas alegres".

Os tipos 2 e 3 são chamados de recuperação semântica de imagens, e o intervalo entre os tipos 1 e 2 é chamado de descontinuidade semântica (*semantic gap*). Mais es-

pecificamente, a descontinuidade semântica é a discrepância entre o poder limitado das características primitivas extraídas das imagens e a riqueza de detalhes da semântica dos usuários. As técnicas conhecidas como realimentação de relevância podem ser empregadas para diminuir essa descontinuidade. Os conceitos referentes a essas técnicas são apresentados no Capítulo 3. Há propostas de sistemas e métodos que tratam de consultas dos tipos 2 e 3 (semânticas), porém essas propostas estão longe de serem consideradas confiáveis [Müller et al., 2004]. A seguir são apresentadas as técnicas de extração de características primitivas.

2.1.1 Extração de características primitivas

Muitos algoritmos sofisticados de extração de características primitivas têm sido desenvolvidos. Diversas pesquisas na área de recuperação por conteúdo indicam que quanto mais especializada for a aplicação de um extrator de características para um domínio de imagens, menor será a descontinuidade semântica gerada. A seguir são apresentadas as principais técnicas de extração de características primitivas para recuperação por conteúdo de imagens encontradas nos sistemas existentes na literatura.

2.1.1.1 Segmentação de Imagens

A segmentação consiste na divisão de uma imagem em partes, de acordo com algum critério ou necessidade. Durante a realização de consultas por conteúdo, geralmente os usuários estão interessados em regiões específicas ao invés das imagens inteiras. Isso acontece uma vez que a representação de imagens baseada em regiões aproxima-se mais da percepção humana do que a representação das imagens inteiras [Jing et al., 2003]. Logo, a segmentação de imagens em processos de busca por conteúdo de imagens pode ser um passo inicial, seguido da extração de características primitivas das regiões segmentadas. Entretanto, a segmentação automática é uma tarefa difícil. Diversas técnicas têm sido propostas, tanto voltadas para domínios de imagens de uso geral como *curve evolution* [Feng et al., 2001] e *graph partitioning* [Shi e Malik, 2000], quanto voltadas a aplicações específicas, como no exemplo apresentado na Figura 2.3 [Balan et al., 2005] para imagens de exames de ressonância magnética que utiliza uma variação do método EM/MPM [Comer e Delp, 2000] para a segmentação das imagens baseado em textura.

Os domínios de imagens reais são em geral ricos tanto em cores quanto em texturas, portanto essas características são úteis para serem usadas para a segmentação. A identificação de texturas é uma das principais dificuldades para os métodos de segmentação, sendo que muitos dos algoritmos propostos requerem uma estimativa dos parâmetros do modelo de texturas presentes nas imagens. A Figura 2.4 apresenta um exemplo de ima-

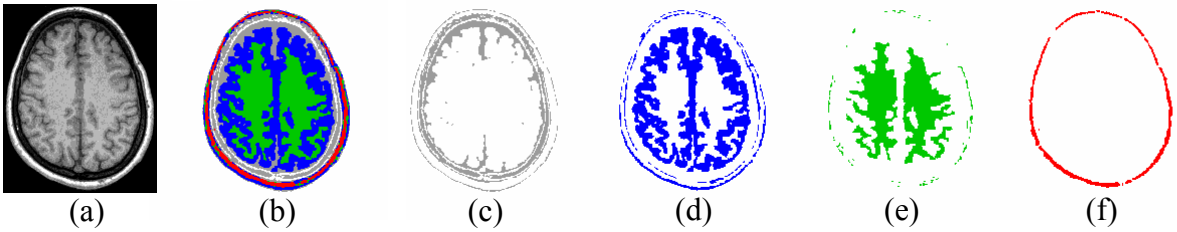

Figura 2.3: Imagem de ressonância magnética e segmentação em quatro classes [Balan et al., 2005]. (a) Imagem original. (b) Imagem segmentada. (c) Classe 1. (d) Classe 2. (e) Classe 3. (f) Classe 4.

gem segmentada automaticamente com base na textura utilizando-se o algoritmo JSEG [Deng e Manjunath, 2001], que se baseia em testes de homogeneidade combinada de padrões de cor e textura.

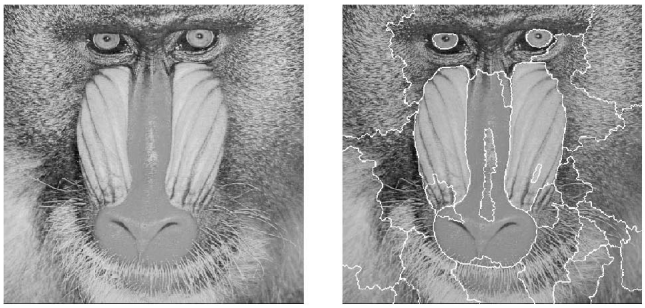

Figura 2.4: Exemplo de segmentação automática baseada em textura. (a) Imagem original. (b) Imagem segmentada [Deng e Manjunath, 2001].

Outra técnica bastante utilizada é a *blobworld segmentation* [Carson et al., 2002]. Nela, a segmentação é obtida pelo agrupamento dos *pixels* em um espaço de posições das cores e texturas, e utiliza o algoritmo *expectation maximization* (EM) para estimar os parâmetros do modelo.

2.1.1.2 Cor

As características baseadas em cores são as mais utilizadas em recuperação por conteúdo, principalmente por apresentarem custo computacional reduzido. O histograma de cores para recuperação de imagens por conteúdo foi introduzido por [Swain e Ballard, 1991]. Ele é obtido pela quantização do espaço de cores e pela contagem do número de *pixels* que cada cor quantizada possui na imagem e, em geral, o resultado é normalizado para evitar diferenças de escala das imagens. A Figura 2.5 apresenta um exemplo de histograma de uma imagem quantizada em 256 níveis de cinza. As vantagens de utilizar histogramas normalizados de cores estão na eficiência em termos da sua computação e em termos de comparação entre histogramas, além de serem invariantes a rotações e trans-

lações das imagens. Histogramas de níveis de cinza podem ser extraídos diretamente de imagens coloridas, de modo que é possível considerar também a iluminação e a saturação [Grundland e Dodgson, 2007]. Como apresentam uma capacidade de discriminação reduzida, eles são normalmente empregados como filtros, que reduzem a quantidade de imagens a serem enviadas a processos mais custosos, baseados em outras características das imagens.

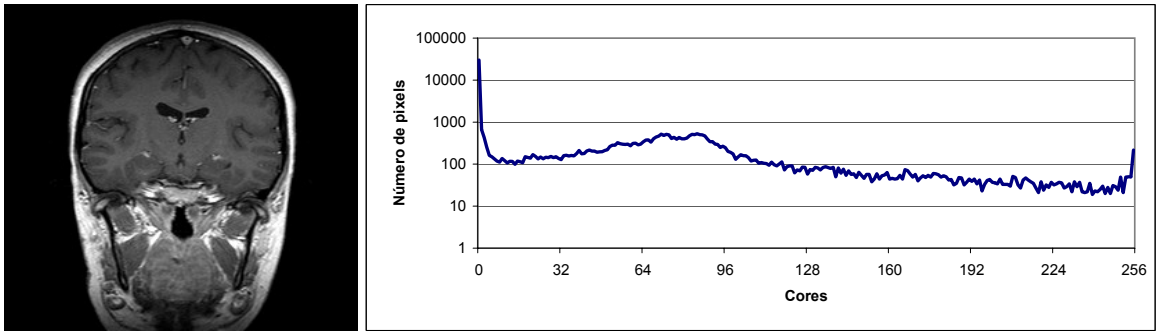

Figura 2.5: Exemplo de histograma. (a) Imagem de ressonância magnética de crânio, quantizada em 256 níveis de cinza. (b) Histograma de níveis de cinza da imagem.

Entretanto, uma desvantagem do histograma de cores é o fato de não apresentar informação sobre a distribuição espacial das cores. Diversas técnicas foram propostas baseadas no histograma de cores para tratar esse problema, entre elas *color coherence vector* [Pass et al., 1996], *color correlogram* [Huang et al., 1997] e *color distribution entropy* [Sun et al., 2006]. Outra desvantagem está no espaço de memória requerido para seu armazenamento. Para reduzir esse problema foram propostos os métodos histograma métrico [Traina et al., 2003] e *cell histogram* [Stehling et al., 2003].

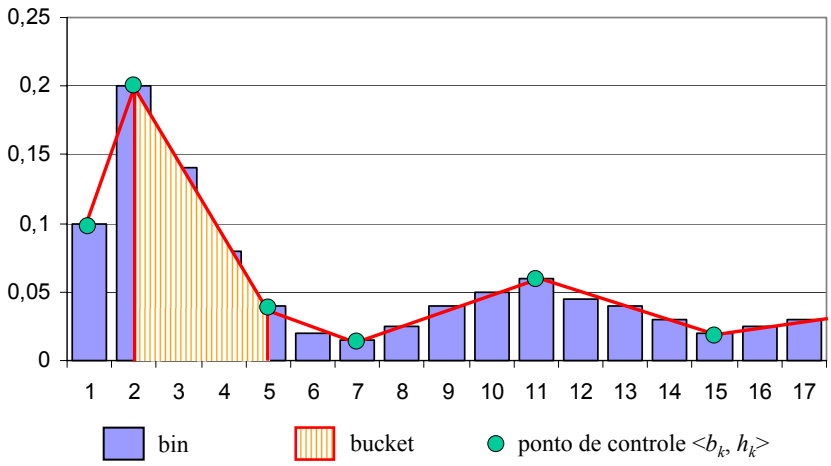

Figura 2.6: Histograma normalizado de níveis de cinza com pontos de controle $< b_k, h_k >$ que definem os *buckets* correspondentes ao seu histograma métrico.

O método histograma métrico reduz a dimensionalidade do vetor de características resultante de um histograma de níveis de cinza por meio da aproximação linear dos valores

(*bins*) do histograma com uma curva, como apresentado na Figura 2.6. Um histograma métrico é formado por um conjunto de *buckets* compostos de pares $< b_k, h_k >$ consecutivos, sendo que b_k indica a largura e h_k a altura do *bucket*. Nele, um conjunto de valores do histograma de níveis de cinza representa um *bucket*, por meio da aproximação linear de uma seqüência de valores do histograma. Os *buckets* não precisam ser regularmente espaçados, e o número de *buckets* em um histograma métrico depende do limiar de aceitação no processo de aproximação linear da curva. O histograma métrico melhora a eficiência de estruturas de indexação para recuperação por conteúdo de imagens em relação ao histograma de níveis de cinza normalizado, uma vez que o espaço exigido para o seu armazenamento é menor.

2.1.1.3 Textura

Textura pode ser definida como "o modo como uma pessoa sente uma superfície ou material ao tocá-la, especialmente quanto à maciez ou rugosidade da mesma" [Longman Dictionary, 2003]. Aplicado a imagens, o termo designa como ocorrem as variações de intensidade de cor, formadas por elementos visíveis arranjados de modo equânime com densidades variadas, nas quais um elemento corresponde a uma região de intensidade uniforme que se repete dentro de um intervalo [Rui et al., 1999]. A Figura 2.7 apresenta três regiões de interesse de imagens de mamografia, com diferentes texturas. É importante notar que a textura pode prover informações importantes acerca da classificação da imagem, uma vez que descreve o conteúdo de muitas imagens reais, sendo considerado um tipo de característica importante para definir o conteúdo de uma imagem [Liu et al., 2007].

Figura 2.7: Exemplos de textura. Regiões de interesse de imagens de mamografia.

Entre as técnicas para extração de características de textura estão os filtros de Gabor [Santini e Jain, 1996] e as transformadas de *wavelets* [Daubechies, 1990, Santini e Gupta, 2001, Arivazhagan e Ganesan, 2003]. Esses métodos tentam capturar partes da imagem com relação à mudança de direção e escala, e são muito úteis para imagens ou regiões com texturas homogêneas. Outras técnicas importantes são as matri-

zes de co-ocorrência [Haralick et al., 1973, Haralick, 1979] e as transformadas de Fourier [Milanese e Cherbuliez, 1999].

2.1.1.4 Forma

A segmentação automática de imagens ainda é um problema que merece grande atenção da comunidade de processamento de imagens. Mesmo em domínios especializados, as técnicas de segmentação ainda causam problemas que muitas vezes não são contornáveis [Müller et al., 2004].

Após a segmentação, os segmentos resultantes podem ser descritos como vetores de características de forma, que podem ter dimensão fixa (momentos) ou variável (coordenadas polares ou aproximação linear de segmentos do contorno para obtenção de coordenadas cartesianas). Vários métodos foram propostos, como os momentos de Zernike [Gu et al., 2002, Bin e Jia-Xiong, 2002, Kotoulas e Andreadis, 2005] e o método *curvature scale space* [Manjunath et al., 2002, Mokhtarian e Abbasi, 2002], que tem como principal característica ser invariante a translações, rotações e escalas. A Figura 2.8 apresenta exemplos de imagens de tomografia computadorizada e a extração de descritores de forma obtidos automaticamente com o uso do método denominado *variational level set* [Li et al., 2006].

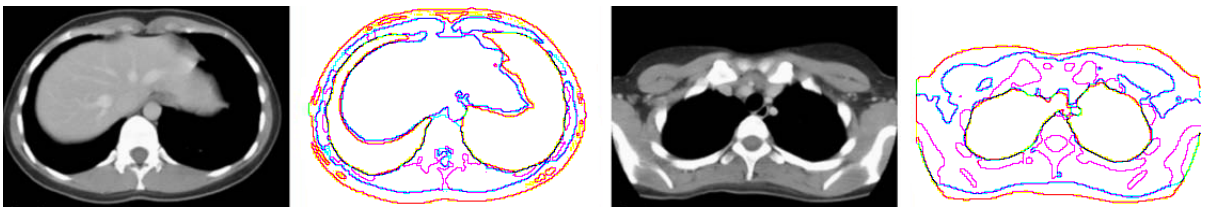

Figura 2.8: Exemplos de extração automática de descritores de forma de imagens de tomografia computadorizada [Li et al., 2006].

2.1.2 Métodos Específicos para Imagens de Exames Médicos

Há diversas técnicas desenvolvidas para domínios específicos de imagens de exames médicos. Um domínio de imagens que tem atraído grande atenção da comunidade científica, dada sua importância na prevenção e auxílio ao diagnóstico de câncer, é o de mamografia de rotina (também conhecida como *screening*). As imagens são resultados de raios-x de baixa intensidade, captadas em aparelho específico. Em [Ferrari et al., 2004] e [Alto et al., 2005] são apresentados métodos para identificação de nódulos utilizando respectivamente *wavelets* de Gabor e características de textura de Haralick. Em [Qian et al., 2002] é utilizada a segmentação para identificação de grupos de microcalcificações e em [Soltanian-Zadeh et al., 2004] também são utilizadas *wavelets* para extração

de características de textura para classificação de microcalcificações. Em [Wei et al., 2009] técnicas de extração de forma são utilizadas para detectar agrupamentos de microcalcificações e em [Dua et al., 2009] características de textura são utilizadas para treinar um classificador para auxílio ao diagnóstico.

Em [Schilham et al., 2006] é utilizada a segmentação de imagens de raios-x de tórax para detecção de nódulos pulmonares para auxílio ao diagnóstico de câncer de pulmão e em [Dy et al., 2003] características de cor, textura e forma são combinadas em um mesmo vetor para serem usadas na recuperação de imagens de tomografia computadorizada de tórax. Em [Antani et al., 2006] são utilizados imagens de raios-x da coluna vertebral para auxílio ao diagnóstico de degeneração de vértebras.

Com relação às imagens de ressonância magnética, que apresentam grande quantidade e qualidade de informação visual, há interesse de diversas áreas na medicina. Por exemplo, em [Siadat e Soltanian-Zadeh, 2005] é apresentado um protótipo que utiliza segmentação para identificação e análise do hipocampo juntamente com informações textuais de laudos para identificação de correlações nos exames para o auxílio ao diagnóstico de epilepsia.

É importante notar que, para o auxílio ao diagnóstico baseado em imagens de exames, devem ser desenvolvidas técnicas específicas. Muitas dessas técnicas ainda estão em desenvolvimento, e há necessidade de mais pesquisas nessa área.

2.2 Medidas de Similaridade

A recuperação por conteúdo baseia-se no cálculo da dissimilaridade entre imagens, de modo que uma imagem de consulta é comparada a uma base de imagens. Nesse modelo, o resultado de uma busca é uma lista de imagens ordenada pela dissimilaridade das mesmas em relação à imagem de consulta. A dissimilaridade entre duas imagens é um valor real positivo, resultado da aplicação de uma função de distância, e quanto menor o seu valor, menor será a dissimilaridade entre elas. As funções de distância apresentam um papel importante nas buscas por conteúdo, de modo que diferentes funções de distância podem retornar imagens diferentes como resultado. O modelo matemático que descreve uma função de distância é chamado espaço métrico [Jacobs et al., 2000, Kutz et al., 2003]. Um espaço métrico é um par $< W, d >$, sendo que W é um domínio de elementos e d é uma função de distância, denominada métrica, que satisfaz os seguintes axiomas para qualquer elemento $x, y, z \in W$:

- Identidade: $d(x, x) = 0$
- Simetria: $d(x, y) = d(y, x)$
- Não-negatividade: $0 \leq d(x, y) < \infty$

- Desigualdade triangular: $d(x, y) \leq d(x, z) + d(z, y)$

Um espaço multidimensional pode gerar um espaço métrico se o mesmo for associado a uma métrica. Entre as métricas amplamente utilizadas estão aquelas da família Minkowski ou métricas L_p, que são aplicadas em domínios multidimensionais. Essas métricas são definidas pela Equação 2.1 pela variação do parâmetro $p \in \mathbb{R} \mid p \geq 1$, sendo que x e y são vetores e n é o número de dimensões dos vetores. Suas principais funções de distância são: Manhattan ou *City Block* ($p = 1$), Euclidiana ($p = 2$) e Chebychev ($p = \infty$). A ilustração da abrangência dessas funções em um espaço bidimensional é apresentada na Figura 2.9.

$$d(x, y) = \sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p} \tag{2.1}$$

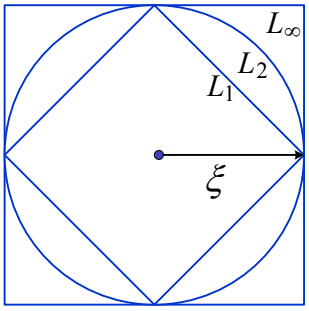

Figura 2.9: Abrangência das funções L_1, L_2 e L_∞ em um espaço bidimensional.

Outra função de distância comumente utilizada é a distância de Mahalanobis, introduzida pelo estatístico Prasanta Chandra Mahalanobis em 1936, que considera a relação de covariância entre os atributos. Para tanto, é computada a matriz de covariância V do conjunto, que é utilizada pela função de distância para o cálculo da dissimilaridade entre os vetores x e y, conforme Equação 2.2.

$$d(x, y) = \sqrt{(x - y)^T \cdot V^{-1} \cdot (x - y)} \tag{2.2}$$

Já a função de distância Canberra, definida na Equação 2.3, pode ser aplicada apenas em vetores com valores não negativos. Essa função é bastante sensível a pequenas mudanças entre os valores de uma coordenada quando os seus valores em ambos os vetores são próximos de zero. Seu comportamento é semelhante ao da distância Manhattan, pois se baseia no cálculo das diferenças absolutas de cada dimensão. Na equação, x e y são vetores, n é a dimensão dos mesmos e se ambos os valores de uma coordenada são iguais a zero, assume-se que $0/0 = 0$. Em [Androutsos et al., 1998], a função de distância Canberra foi utilizada para recuperação de imagens baseada em histogramas de cores. É importante notar que, mesmo que mais de uma métrica possa tecnicamente ser utili-

zada para comparar vetores de características obtidos por um determinado extrator, cada métrica leva a uma precisão diferente [Bugatti et al., 2008].

$$d(x,y) = \sum_{i=1}^{n} \frac{|x_i - y_i|}{|x_i + y_i|} \tag{2.3}$$

As características extraídas, em particular as que não têm dimensão fixa, necessitam do desenvolvimento de funções de distância específicas. Por exemplo, no caso descrito anteriormente na Seção 2.1.1.2, cada vetor que representa um histograma métrico pode ter uma quantidade definida de *buckets* e cada *bucket* pode ter sua largura diferente dos demais *buckets*. A função de distância que compara dois histogramas métricos A e B é definida pela Equação 2.4, que corresponde à área entre as curvas que representam dois histogramas métricos [Traina et al., 2003]. A Figura 2.10 apresenta dois histogramas métricos com seus respectivos pontos de controle $< b_k, h_k >$. É importante notar que essa função de distância atende às propriedades de um espaço métrico.

$$d_{MH}(A,B) = \int_{x=0}^{n} |A_{<b_x,h_x>} - B_{<b_x,h_x>}|\, dx \tag{2.4}$$

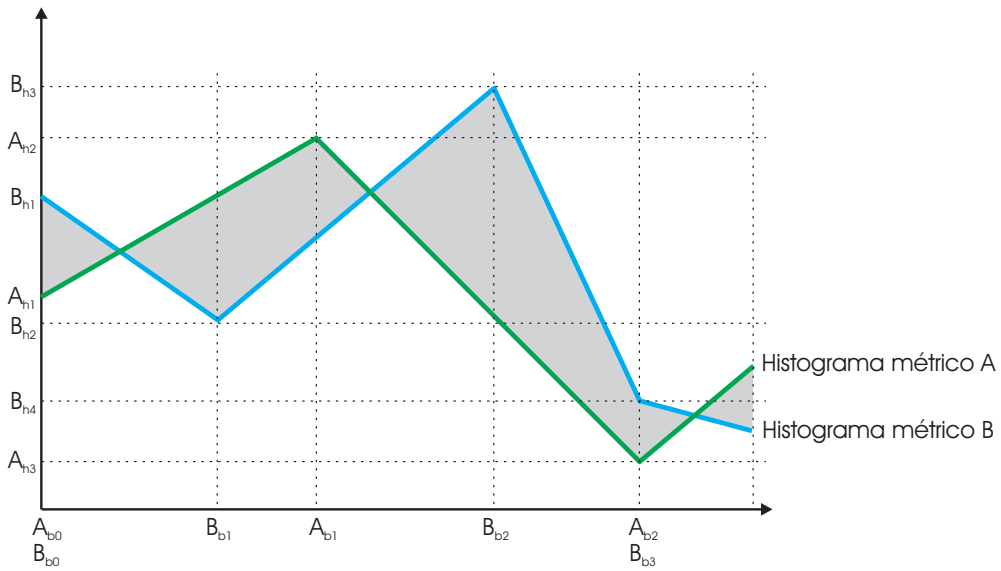

Figura 2.10: Distância entre dois histogramas métricos A e B calculada pela área definida pelos pontos de controle $< b_k, h_k >$.

Em se tratando de cadeias de caracteres, é possível computar a distância entre duas cadeias com a distância de edição (*edit distance*), sendo seu resultado o número mínimo de operações de inserção, remoção ou substituição de caracteres necessários para transformar uma cadeia em outra. Um custo é atribuído à cada operação, e a distância entre duas cadeias de caracteres é a soma desses custos. A distância *Levenshtein* é a distância de edição com todos os custos iguais a 1 [Euzenat e Shvaiko, 2007].

Considerando documentos de texto, a técnica de freqüência do termo (*term frequency – TF*) é comumente usada para recuperação de informações. Ela baseia-se na computação da relevância de cada termo para um documento, sendo que sua relevância aumenta proporcionalmente ao número de vezes que ela aparece no mesmo. Nesse modelo, cada documento de texto é representado por um vetor em um espaço de alta dimensão, de modo que a freqüência de cada termo é computada pela Equação 2.5, resultando em uma coordenada para cada termo. Como o número de termos de uma coleção de documentos tende a ser grande e o número de termos de um documento em relação ao conjunto total de termos tende a ser proporcionalmente pequeno, o vetor que representa um documento tende a ser esparso, logo sua representação pode ser dada por uma estrutura composta por pares contendo a posição do termo presente no vetor e o valor da sua freqüência. Em geral, a similaridade do cosseno é usada para medir a quantidade e relevância dos termos compartilhados entre dois documentos. Apesar da similaridade do cosseno não atender às propriedades de uma função de distância, o inverso da função cosseno permite a computação de distância entre os vetores. Assim, a distância entre um elemento de consulta q e um documento e é dada pela correlação entre os vetores $\vec{e}$ e $\vec{q}$, computada pelo arco-cosseno do ângulo entre os dois vetores, conforme a Equação 2.6 [Baeza-Yates e Ribeiro-Neto, 1999].

$$\text{freqüência do termo} = \frac{\text{número de ocorrências do termo no documento}}{\text{número de ocorrências de todos os termos do documento}} \tag{2.5}$$

$$d(e, q) = \text{arco-cosseno}(\vec{e} \cdot \vec{q}) \tag{2.6}$$

Garantidas as propriedades do espaço métrico, é possível realizar consultas por similaridade em grandes bases de imagens de modo eficiente. Entretanto, há alguns trabalhos que utilizam funções para o cálculo de dissimilaridade que ferem o axioma da desigualdade triangular, com o objetivo de aproximar a dissimilaridade computada do resultado esperado por especialistas de um domínio, entre eles [Jacobs et al., 2000, Qamra et al., 2005]. Isso ocorre porque tem sido reconhecido que a percepção humana de similaridade tende a não seguir as propriedades de uma métrica [Felipe et al., 2009]. Apesar desses trabalhos apresentarem resultados promissores, até o momento eles não são escaláveis para grandes conjuntos de dados.

2.3 Consultas por Similaridade

Em geral, as aplicações tradicionais que manipulam dados numéricos e textuais realizam consultas baseadas em igualdade e ordem total. Para os dados de natureza multimídia,

consultas baseadas na dissimilaridade são desejadas. Após a realização da extração de características das imagens de um conjunto e da escolha de uma função de distância apropriada, as características extraídas passam a representar cada imagem como um elemento em um espaço métrico definido pelo domínio das características e por uma função de distância apropriada, definida sobre o conjunto de vetores de características. Uma consulta por similaridade é geralmente definida por um elemento de consulta e uma restrição baseada na proximidade (distância) em relação ao elemento de consulta. A seguir são apresentados os principais tipos de consultas por similaridade [Zezula et al., 2006].

2.3.1 Consultas por Abrangência

Seja $\mathbb{S}$ um domínio de dados. Uma consulta por abrangência (*range query*) recupera todo elemento e de um conjunto de dados $S \subseteq \mathbb{S}$ que se encontra a até uma distância máxima ξ (raio de busca) do elemento de consulta $q \in \mathbb{S}$. Formalmente:

$$Rq(q, \xi) = \{e \in S \mid d(e, q) \leq \xi\}$$

Opcionalmente, os elementos do resultado podem ser retornados ordenados em relação à distância do elemento de consulta q. É importante notar que o elemento q não precisa fazer parte da coleção de elementos que serão consultados, porém ele deve pertencer ao domínio métrico que define o conjunto de dados S. Quando o raio de consulta $\xi = 0$, a consulta por abrangência é chamada consulta pontual (*point query* ou *exact match*). A Figura 2.11 apresenta uma ilustração da consulta por abrangência em um espaço euclidiano bidimensional com raio de busca ξ.

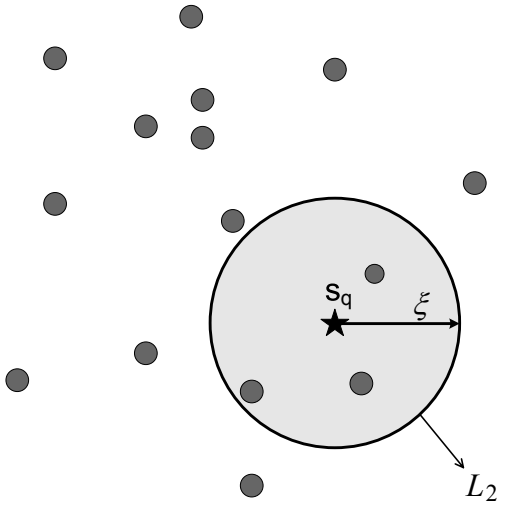

Figura 2.11: Consulta por abrangência com centro de consulta s_q e raio de busca ξ.

2.3.2 Consultas aos k-Vizinhos mais Próximos

Muitas vezes é difícil determinar um raio de busca ξ sem um prévio conhecimento da distribuição do conjunto de dados e da função de distância. Um modo de limitar uma consulta por similaridade é informar um valor k de elementos mais próximos desejados. Uma consulta aos k-vizinhos mais próximos (*k-nearest neighbor query*) recupera os k elementos do conjunto de dados $S \subseteq \mathbb{S}$ mais próximos ao elemento de consulta $q \in \mathbb{S}$. Formalmente:

$$NNq(q,k) = \{R \subseteq S, |R| = k \wedge \forall x \in R, y \in S - R : d(q,x) \leq d(q,y)\}$$

Em [Ilyas et al., 2008] é apresentado um apanhado de técnicas relacionadas às consultas aos k-vizinhos mais próximos. A Figura 2.12 apresenta uma ilustração da consulta por vizinhos mais próximos em um espaço euclidiano bidimensional com $k = 4$.

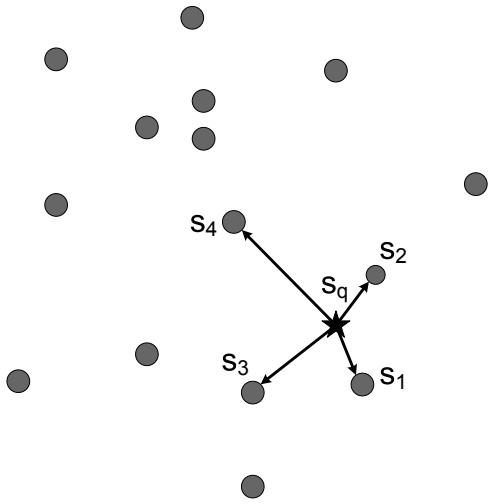

Figura 2.12: Consulta aos k-vizinhos mais próximos a partir do centro de consulta s_q e $k = 4$.

2.3.3 Consultas por Similaridade Baseadas em Múltiplos Centros

As consultas por abrangência e aos k-vizinhos mais próximos consideram apenas um elemento de consulta. Entretanto, diversas aplicações necessitam computar a dissimilaridade dos elementos de um conjunto de dados para mais de um elemento de consulta. Nesse caso, uma função que considera a dissimilaridade de um elemento para um conjunto de elementos deve ser empregada. As consultas por similaridade com múltiplos centros baseiam-se em funções de agregação para computar a distância de um elemento para o conjunto de

centros de consulta. Considerando espaços métricos, Wu *et al.* [Wu et al., 2000] propuseram a técnica Falcon para ordenar imagens baseada em uma função de agregação de distâncias e um determinado raio de consulta ξ, na qual a agregação das distâncias de um elemento para os centros de consultas é dada pela raiz α do somatório das distâncias elevadas a potência α. O trabalho apresentado em [Papadias et al., 2005] propõe o uso de uma função de agregação para ordenar dados multidimensionais de baixa dimensionalidade para resolver uma consulta aos k-vizinhos mais próximos de um conjunto de centros de consulta. O trabalho propõe o uso das funções de agregação soma, mínimo e máximo. A Figura 2.13 apresenta uma ilustração das consultas por abrangência agregada e aos vizinhos mais próximos agregado em um espaço euclidiano bidimensional considerando 3 centros de consulta.

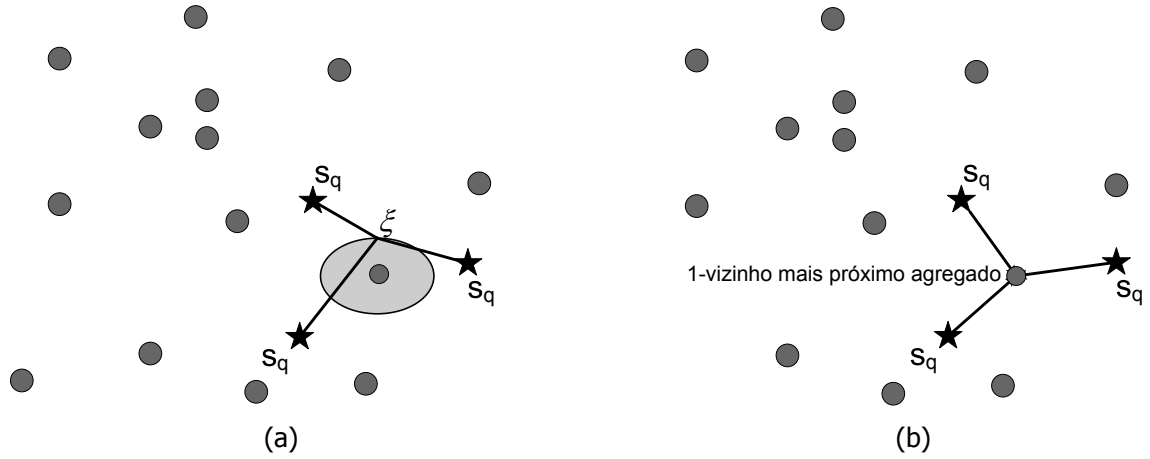

Figura 2.13: Ilustração do uso de agregação de distâncias em consultas por similaridade. (a) Consulta por abrangência agregada. (b) Consulta aos k-vizinhos mais próximos agregado.

O trabalho apresentado em [Tahaghoghi et al., 2002] também propõe o uso das funções de agregação soma, mínimo e máximo para ordenar características extraídas de imagens e apresenta uma avaliação empírica dessas funções.

2.4 Estruturas de Indexação para Consultas por Similaridade

Para a realização eficiente de consultas por similaridade, é necessário armazenar as características extraídas em métodos de acesso apropriados para a realização de consultas por abrangência e aos k-vizinhos mais próximos. Esses métodos de acesso devem resolver com eficiência ao menos os seguintes tipos de consultas [Böhm et al., 2001]:

- consultas pontuais;

- consultas por abrangência;
- consultas aos k-vizinhos mais próximos.

Entre os principais métodos de acesso para dados multidimensionais destacam-se os métodos baseados na *R-tree* [Guttman, 1984, Sellis et al., 1987, Beckmann et al., 1990] e para dados em domínios métricos destacam-se os métodos *M-tree* [Ciaccia et al., 1997] e *Slim-tree* [Traina-Jr et al., 2002], descritos a seguir. Um espaço multidimensional é um espaço métrico quando é fornecida uma função de distância.

2.4.1 Métodos de Acesso Multidimensionais

O objetivo dos métodos de acesso multidimensionais (*multidimensional access methods*) é a manipulação de dados multidimensionais e geométricos como pontos, segmentos de reta, planos, volumes e hipervolumes em espaços de alta dimensão. O método R-tree proposto por Guttman [Guttman, 1984] é uma estrutura hierárquica baseada no método B^+-tree que permite a organização dinâmica de um conjunto de elementos geométricos de dimensão d pela representação de retângulos envolventes mínimos (*minimum bounding rectangle – MBR*). Cada nodo de uma R-tree corresponde a um MBR que envolve seus descendentes, sendo que pode haver sobreposição entre diversos MBR. A Figura 2.14 apresenta um conjunto de MBR com sua distribuição espacial e respectiva estrutura lógica, assumindo uma capacidade máxima de 4 MBR por nodo [Manolopoulos et al., 2005].

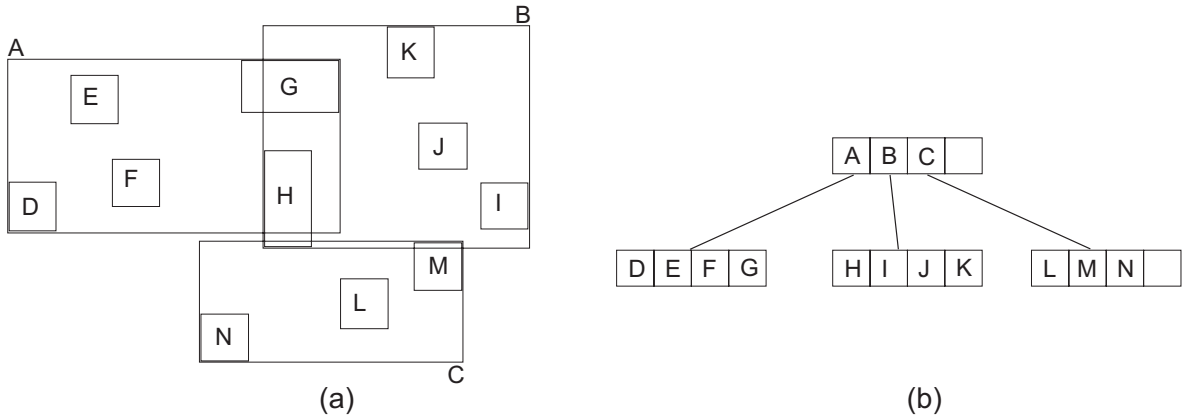

Figura 2.14: Representação de uma R-tree com 14 MBR [Manolopoulos et al., 2005]. (a) Distribuição espacial. (b) Estrutura lógica.

A Figura 2.15 apresenta os conjuntos de MBR de um índice R-tree para o conjunto de dados de coordenadas geográficas das cidades brasileiras, que resultou em um índice de altura igual a 4.

É importante notar que a R-tree permite a realização de diversos tipos de consultas, dentre elas, topológicas, direcionais, categóricas e baseadas em distância, abrangendo

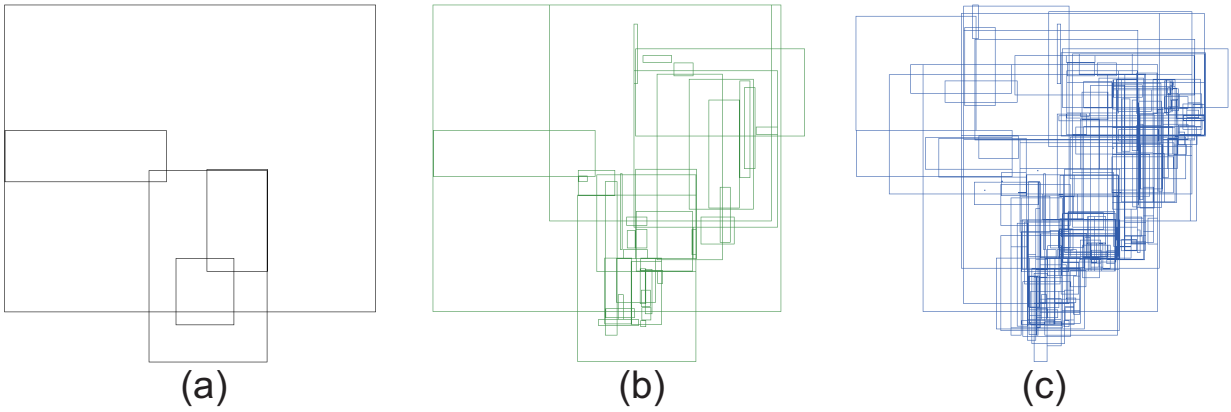

Figura 2.15: MBR de um índice R-tree para o conjunto de coordenadas geográficas das cidades brasileiras. (a) Conjunto de MBR do primeiro nível do índice. (b) Segundo nível do índice. (c) Terceiro nível do índice.

aplicações multimídia, de *data warehousing* e de mineração de dados. Entretanto, o desempenho das R-tree e variantes degrada com o aumento no número de dimensões do conjunto de dados, o que inviabiliza seu uso em recuperação de imagens por conteúdo, dado que nesses domínios os vetores de características usualmente são compostos por um grande número de dimensões. Uma revisão dos métodos de acesso multidimensionais pode ser encontrada em [Gaede e Günther, 1998, Ahn et al., 2001, Samet, 2006] e em particular para os métodos baseados na R-tree pode ser encontrada em [Manolopoulos et al., 2005].

2.4.2 Métodos de Acesso Métricos

Os principais objetivos das estruturas de indexação para espaços métricos, denominadas métodos de acesso métrico (*metric access methods – MAM*), são a redução do número de cálculos de distância e a redução do número de acessos a disco para realização de consultas baseadas em distância (consultas por similaridade). Em pouco mais de uma década, diversos trabalhos foram propostos para a criação de estruturas eficientes, dentre eles a VP-tree (*Vantage Point tree*) [Yianilos, 1993], a MVP-tree (*Multi-Vantage Point tree*) [Bozkaya e Özsoyoglu, 1997], a M-tree [Ciaccia et al., 1997] e a Slim-tree [Traina-Jr et al., 2002], sendo que a M-tree e a Slim-tree são as primeiras estruturas dinâmicas balanceadas. Uma revisão desses métodos pode ser encontrada em [Zezula et al., 2006, Samet, 2006].

A Slim-tree é um MAM que cresce a partir das folhas em direção à raiz (*bottom-up*). Assim como outros MAM, ela agrupa os elementos em páginas de tamanho fixo, sendo que cada página corresponde a um nó da árvore. A Figura 2.16 representa uma Slim-tree com três níveis em um espaço de duas dimensões, sendo que os círculos brancos são os nós folha e os círculos em cinza são os nós índice. Cada nó da árvore possui um elemento representante s_{rep} (círculos pretos) e um raio de cobertura do nó.

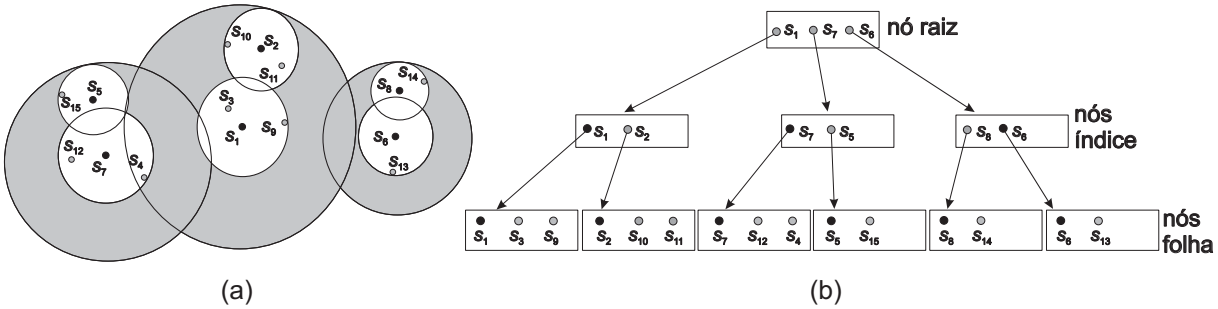

Figura 2.16: Representação de uma Slim-tree com 15 elementos organizados em 3 níveis e com capacidade máxima do nó igual a 3. (a) Distribuição espacial. (b) Estrutura lógica.

2.4.3 Otimização de Consultas por Similaridade

A propriedade de desigualdade triangular é usada durante uma consulta por similaridade para a poda de elementos e sub-árvores que, com certeza, não fazem parte do conjunto resposta. A vantagem é que com a poda não é preciso calcular as distâncias entre o elemento de consulta e todos os elementos armazenados na sub-árvore, caso não haja sobreposição entre a cobertura do nodo e a cobertura da consulta. Dessa maneira, a quantidade de cálculos de distância e de acessos a disco em uma consulta pode ser reduzida, proporcionando melhor desempenho para responder às consultas. Dado um elemento de consulta s_q, um raio de consulta ξ, um elemento representante s_{rep} de um nodo e seu respectivo raio de cobertura ξ_{nodo}, a maneira como pode ser feito o descarte usando a propriedade da desigualdade triangular pode ser ilustrada pela Figura 2.17. O nodo representado por s_{rep} e ξ_{nodo} poderá ser descartado se: $d(s_{rep}, s_q) > \xi_{nodo} + \xi$.

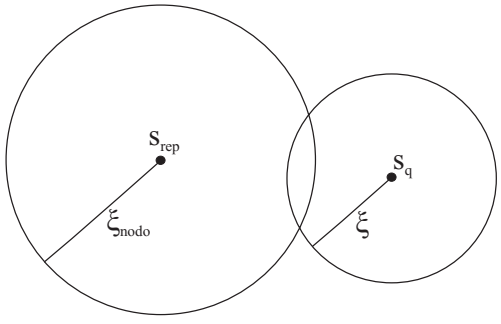

Figura 2.17: Descarte pela desigualdade triangular. Os elementos ou sub-árvores que estiverem no nodo representado por s_{rep} poderão ser descartados se $d(s_{rep}, s_q) > \xi_{nodo} + \xi$.

As propriedades dos espaços métricos são essenciais para determinar se há intersecção entre duas regiões em um determinado espaço. Considere o contra-exemplo no qual é demonstrado como a utilização de $p < 1$ na Equação 2.1 (Minkowski) não define uma função de distância. A Figura 2.18 apresenta um exemplo de espaço definido pela função $L_{0,5}$ e pelos elementos $a = (1,0\ ,\ 1,0)$, $b = (1,8\ ,\ 1,4)$ e $c = (1,7\ ,\ 1,0)$. Os elementos a e b definem regiões, por meio dos raios de cobertura $r_a = 1,0$ e $r_b = 1,0$ definidos por

uma função de distância. A aplicação da função $L_{0,5}$ resulta nas medidas $d(a, b) = 2, 1$, $d(a, c) = 0, 8$ e $d(b, c) = 1, 1$. Como a propriedade de desigualdade triangular não é válida, uma vez que $d(a, b) > d(a, c) + d(b, c)$, não é possível determinar com essa propriedade se o elemento c está coberto pela intersecção entre as duas regiões definidas pelos raios r_a e r_b a partir dos elementos a e b respectivamente.

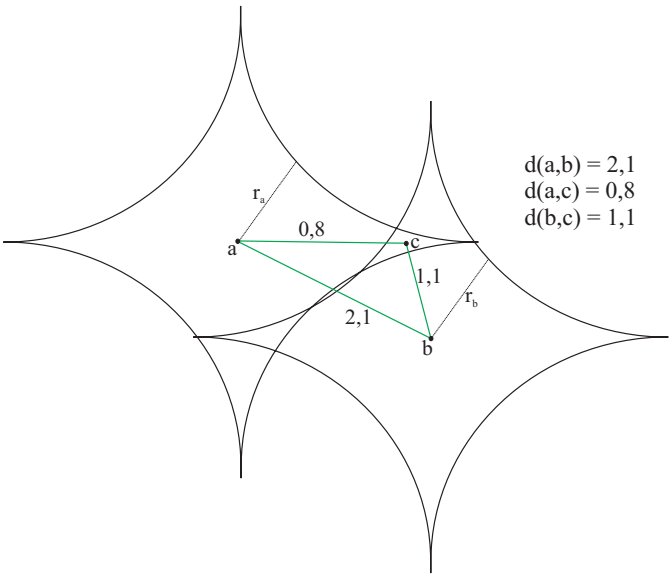

Figura 2.18: Contra-exemplo da utilização da desigualdade triangular para determinar a sobreposição. Considere os elementos $a = (1,0\ ,\ 1,0)$ e $b = (1,8\ ,\ 1,4)$, seus raios correspondentes $r_a = 1,0$ e $r_b = 1,0$, e o elemento $c = (1,7\ ,\ 1,0)$ em um espaço definido pela função $L_{0,5}$. Como a função $L_{0,5}$ em um espaço multidimensional não define um espaço métrico, não é possível utilizar a propriedade de desigualdade triangular para determinar se há intersecção entre as duas regiões.

Nas últimas décadas, várias abordagens para otimização de consultas aos k-vizinhos mais próximos foram propostas, como *branch-and-bound* [Roussopoulos et al., 1995, Ciaccia et al., 1997, Samet, 2003], incremental [Hjaltason e Samet, 2003] e algoritmos *multi-step* [Korn et al., 1996, Seidl e Kriegel, 1998], muitas delas baseadas em métodos de acesso espaciais ou métricos. Outras abordagens tentam estimar o limite do raio final para a consulta [Tasan e Ozsoyoglu, 2004, Vieira et al., 2007]. Esses trabalhos referem-se a algoritmos que lidam com apenas um centro de consulta.

Com relação a múltiplos centros de consulta, a consulta por abrangência agregada foi proposta em [Wu et al., 2000] na técnica denominada Falcon. O algoritmo de busca proposto consiste na união de consultas por abrangência de único centro executadas para cada centro da consulta, seguida de um filtro que avalia se cada elemento da união atende ao critério de similaridade agregada. O método proposto depende de um limiar empírico ϵ dado pelo usuário para limitar cada consulta por abrangência, cuja semântica depende da noção de similaridade do conjunto de dados pelo usuário, e nenhum método foi apresentado para avaliação do ϵ.

Com relação a consultas com múltiplos centros limitadas por k, o trabalho apresentado em [Namnandorj et al., 2008] propõe a otimização para as funções de agregação soma e máximo. Em [Papadias et al., 2005] é proposto o *minimum bounding method* (MBM), que se baseia em uma função de agregação para ordenar dados espaciais de baixa dimensionalidade e computar os k-vizinhos mais próximos agregados. O método usa propriedades geométricas para realizar podas em R*-trees, apenas para as funções de agregação soma, mínimo e máximo em espaços euclidianos, e não pode ser generalizado para dados métricos. A Figura 2.19 apresenta uma ilustração do método R*-tree MBM. O método baseia-se na definição de uma função, denominada *mindist*, capaz de computar a distância entre um elemento de consulta $q_i \in Q$ e um MBR, que seja equivalente à função de distância alvo da consulta.

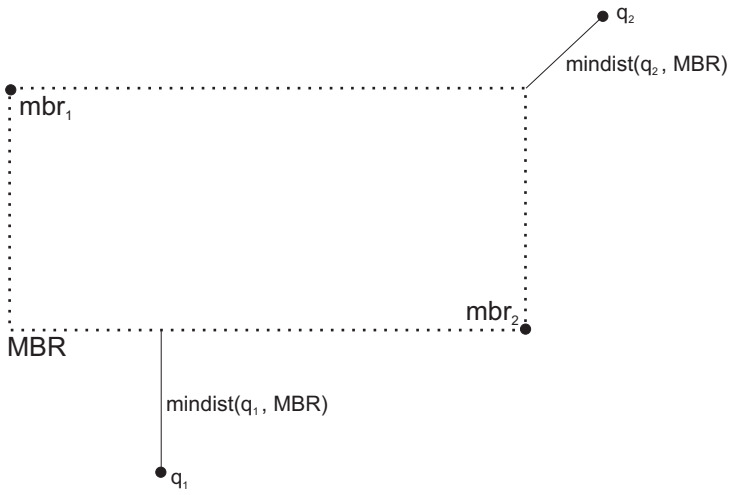

Figura 2.19: Ilustração do método R*-tree MBM. As distâncias entre os elementos de consulta e o MBR definido pelos elementos mbr_1 e mbr_2 são computadas pela função *mindist*.

A partir da raiz do índice, o método armazena na variável *melhorDistância* a menor distância agregada encontrada. A poda de um MBR pode ser realizada para a função de agregação soma se $\sum_{i=1}^{|Q|} mindist(q_i, MBR) > melhorDistância$, para a função de agregação máximo se $max_{i=1}^{|Q|} mindist(q_i, MBR) > melhorDistância$ e para a função de agregação mínimo se $min_{i=1}^{|Q|} mindist(q_i, MBR) > melhorDistância$.

2.4.4 Avaliação da Qualidade dos Resultados de Consultas em CBIR

Provost *et al.* [Provost et al., 1998] provaram que a simples análise da acurácia para validação empírica de algoritmos pode induzir a resultados errôneos e sugeriram o uso de curvas ROC (*receiver operating characteristic – ROC*), que mostram como o número de exemplos classificados corretamente como positivos varia com relação ao número de exem-

plos incorretamente classificados como negativos. Entretanto, as curvas ROC podem apresentar uma visão otimista do desempenho do algoritmo. Assim, muitos pesquisadores têm considerado o uso de curvas de precisão e revocação (*precision and recall – PR*) como uma alternativa às curvas ROC [Baeza-Yates e Ribeiro-Neto, 1999, Davis e Goadrich, 2006].

Em um problema de decisão binária, um classificador rotula exemplos como positivos ou negativos. A decisão feita pelo classificador pode ser representada por uma estrutura conhecida por Matriz de Confusão, apresentada na Tabela 2.1, que tem quatro categorias:

- Verdadeiro positivo (VP): exemplo rotulado corretamente como positivo;
- Falso positivo (FP): exemplo negativo rotulado incorretamente como positivo;
- Verdadeiro negativo (VN): exemplo rotulado corretamente como negativo;
- Falso negativo (FN): exemplo positivo rotulado incorretamente como negativo.

Tabela 2.1: Matriz de confusão

	Exemplo positivo	Exemplo negativo
Classificado como positivo	Verdadeiro positivo	Falso positivo
Classificado como negativo	Falso negativo	Verdadeiro negativo

Uma curva ROC é o resultado da taxa de falsos positivos no eixo x e a taxa de verdadeiros positivos no eixo y, ou seja, a razão entre os exemplos negativos classificados incorretamente e os exemplos positivos classificados corretamente, conforme as Equações 2.7 e 2.8:

$$\text{Taxa de verdadeiros positivos} = \frac{VP}{VP + FN} \tag{2.7}$$

$$\text{Taxa de falsos positivos} = \frac{FP}{FP + VN} \tag{2.8}$$

Uma curva de PR é o resultado da taxa de revocação (eixo x) pela taxa de precisão (eixo y), definido pelas Equações 2.9 e 2.10:

$$\text{Revocação} = \frac{VP}{VP + FN} \tag{2.9}$$

$$\text{Precisão} = \frac{VP}{VP + FP} \tag{2.10}$$

O objetivo de uma curva de PR é alcançar o canto direito superior do gráfico, ou seja, precisão de 100% para uma revocação de 100%. A Figura 2.20 apresenta duas curvas

de PR, representando os métodos A e B. Nesse exemplo, o método B apresenta melhor resultado que o método A, e é possível observar que há um grande espaço para melhora do resultado.

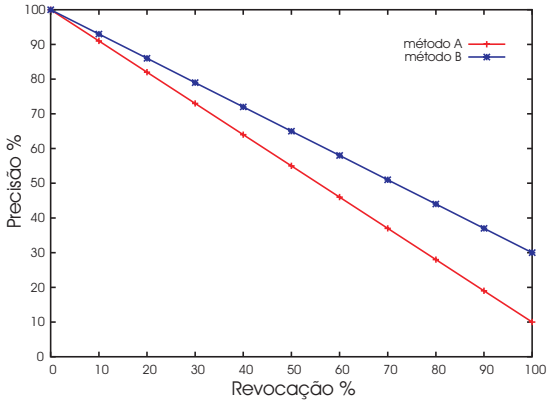

Figura 2.20: Exemplo de gráfico de precisão e revocação.

2.5 Considerações Finais

Este capítulo abordou os conceitos relacionados com os sistemas de recuperação de imagens por conteúdo, tendo como principais enfoques as técnicas de extração de características e a realização de consultas por similaridade. Um sistema de gerenciamento de banco de dados (SGBD) é o ambiente natural para a integração das diversas técnicas envolvidas para a realização de consultas por similaridade de imagens. Porém ainda há várias questões a serem pesquisadas para tornar essas consultas possíveis em SGBD, como a criação de linguagens de consulta para consultas por similaridade de imagens e de outros tipos de dados multimídia, a otimização de planos de consultas, o acesso concorrente a estruturas de indexação métricas, entre outras. Além dessas questões, há a necessidade de desenvolvimento e integração de técnicas de aprendizado de máquina e de realimentação de relevância para diminuir a descontinuidade semântica entre as características de baixo nível que podem ser extraídas automaticamente das imagens e os conceitos semânticos de alto nível. As questões relacionadas ao tratamento da descontinuidade semântica serão apresentadas no próximo capítulo.

Capítulo

3

Abordagens para o Tratamento da Descontinuidade Semântica

Embora muita pesquisa tenha sido realizada na área, a recuperação de imagens por conteúdo ainda é uma questão em aberto. O principal problema está relacionado à descontinuidade semântica entre as características de baixo nível extraídas das imagens e a subjetividade da interpretação humana [Hoi et al., 2006]. As características que podem ser extraídas automaticamente de imagens, mesmo que derivadas de regiões segmentadas das imagens, em geral não correspondem aos conceitos semânticos ou às estruturas que um usuário possa estar interessado. Assim, em uma consulta, a quantidade de respostas interessantes obtidas em relação ao número total de respostas que deveriam ser encontradas, denominado precisão e revocação [Baeza-Yates e Ribeiro-Neto, 1999] tende a ser baixo. Uma visão geral das abordagens para o tratamento da descontinuidade semântica pode ser encontrada nos seguintes trabalhos de revisão da área [Binderberger e Mehrotra, 2003, Zhou e Huang, 2003, Liu et al., 2007, Zhang et al., 2006, Heesch, 2008, Lavrenko, 2009].

Em [LoBue e DeLoache, 2008] são apresentados experimentos nos quais adultos e crianças de 3 a 5 anos reconheceram cobras em imagens com presença de grama mais rapidamente que objetos não ameaçadores como flores e sapos, e sugere que essa capacidade de reconhecimento é inata do ser humano. O entendimento sobre o modo como um indivíduo interpreta uma imagem ainda é uma questão importante que vem sendo estudada, e é possível afirmar que, em muitos domínios, o reconhecimento de imagens pode ser mais complexo que o simples reconhecimento de texturas e formas. Nesse contexto, as técnicas utilizadas para redução da descontinuidade semântica tentam mapear as relações entre

as características de baixo nível extraídas das imagens e a subjetividade da interpretação humana.

Segundo [Liu et al., 2007], as técnicas utilizadas para redução da descontinuidade semântica na recuperação de imagens por conteúdo podem ser separadas em cinco categorias:

- o uso de métodos de aprendizado de máquina para associar características de baixo nível com os conceitos das consultas;
- o uso de técnicas realimentação de relevância para o sistema 'aprender' a intenção do usuário;
- o uso de ontologias para definir conceitos de alto nível para objetos, por meio da definição de um vocabulário de palavras-chaves que são associadas às imagens com base em características de baixo nível;
- a geração de modelos semânticos (*semantic templates*) para auxílio na recuperação por conteúdo;
- a recuperação de imagens da *web* baseada nas evidências de elementos do HTML e em conteúdo visual, por meio da associação do título da imagem, textos próximos à imagem ou *hyperlinks*.

Algumas dessas técnicas foram propostas para imagens de domínio geral, como as utilizadas pelos sistemas de buscas como o Google Images [Jing e Baluja, 2008], Yahoo Images [van Zwol et al., 2008] e o Altavista Image Search [Altavista, 2009]. Em se tratando de buscas baseadas no conteúdo visual de domínios específicos de imagens e na extração automática de características dessas imagens, os métodos baseados em aprendizado de máquina e em realimentação de relevância são os que têm demonstrado os resultados mais promissores para a busca por conteúdo, e são descritos a seguir.

Este capítulo está estruturado do seguinte modo. A Seção 3.1 trata dos métodos baseados em aprendizado de máquina e a Seção 3.2 apresenta as principais técnicas de realimentação de relevância. A Seção 3.3 aborda brevemente os tipos de relevância existentes e a Seção 3.4 apresenta as otimizações desenvolvidas para técnicas de realimentação de relevância. A Seção 3.5 trata da diversidade em consultas por conteúdo de imagens. Finalmente, a Seção 3.7 apresenta as considerações finais do capítulo.

3.1 Métodos Baseados em Aprendizado de Máquina

Muitas vezes, para derivar a semântica das características de alto nível das imagens, é necessário o uso de técnicas formais como as técnicas de aprendizado de máquina supervisionado ou não-supervisionado. No aprendizado supervisionado o objetivo é induzir conceitos de exemplos que estão pré-classificados, ou seja, estão rotulados com um valor de classe conhecido. De acordo com os valores atribuídos à classe, o problema é conhecido como classificação ou regressão, e o processo de aprendizado dá-se pela apresentação de um conjunto de exemplos de treinamento a um indutor. Por outro lado, no aprendizado não supervisionado, a tarefa do algoritmo é agrupar exemplos não rotulados, ou seja, exemplos que não possuem um atributo de classe especificado. Nesse caso, é possível utilizar algoritmos de aprendizado para descobrir padrões nos dados a partir de alguma caracterização de regularidade. A seguir são apresentadas algumas das principais técnicas de aprendizado supervisionado e não-supervisionado para melhorar os processos de recuperação de imagens por conteúdo.

3.1.1 Aprendizado Supervisionado

3.1.1.1 Classificação

Entre os métodos mais utilizados de aprendizado supervisionado para o relacionamento entre os conceitos de alto nível e as características de baixo nível das imagens estão os métodos baseados em *support vector machines* (SVM) e em classificadores Bayesianos.

A técnica SVM [Boser et al., 1992] foi originalmente desenvolvida para classificação binária, porém tem sido utilizada também para reconhecimento de objetos, classificação de textos, e outros problemas. Dado um conjunto de treinamento composto por n vetores pertencentes a duas classes separadas e um conjunto de rótulos, o objetivo é definir um hiper-plano que separe os vetores. Entre os muitos hiper-planos possíveis, o plano separador ótimo é o plano que maximiza a margem, ou seja, a distância entre o hiper-plano e o vetor mais próximo de cada classe. Os *support vectors* são os exemplos de treinamento que estão próximos desse hiper-plano. A Figura 3.1 apresenta um exemplo de SVM. Para aprender múltiplos conceitos para recuperação de imagens, um SVM tem que ser treinado para cada conceito. Um exemplo de uso de SVM para recuperação de imagens pode ser encontrado em [Shi et al., 2004].

Os classificadores Bayesianos também são utilizados no aprendizado supervisionado de imagens para estimar a probabilidade de uma imagem pertencer a uma determinada classe. Eles baseiam-se no teorema de Bayes [Bayes, 1763, Barnard e Bayes, 1958], que permite computar a classe mais provável para um dado exemplo, baseado na distribuição, assu-

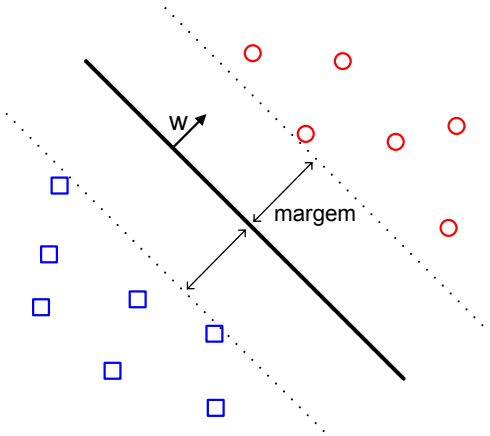

Figura 3.1: Exemplo de SVM linear.

mida como sendo conhecida, do conjunto de dados considerado. Em [Vailaya et al., 2001] eles são utilizados para classificar paisagens baseando-se em características de cores das imagens.

Além dessas técnicas, as árvores de decisão também podem ser usadas para derivar características semânticas, por meio da definição de regras obtidas pela navegação nos caminhos encontrados nas características de baixo nível. A idéia básica desses algoritmos é a realização de um particionamento recursivo do espaço de atributos de entrada até que cada região do espaço possa ser rotulada com o valor de uma única classe, conforme apresentado na Figura 3.2. O C4.5 [Quinlan, 1993] é um dos algoritmos mais utilizados para indução de árvores de decisão. Em [MacArthur et al., 2000] o algoritmo C4.5 é utilizado para construir uma árvore de decisão sobre imagens definidas como relevantes, que é usada como modelo de classificação de imagens em duas classes (relevante e irrelevante).

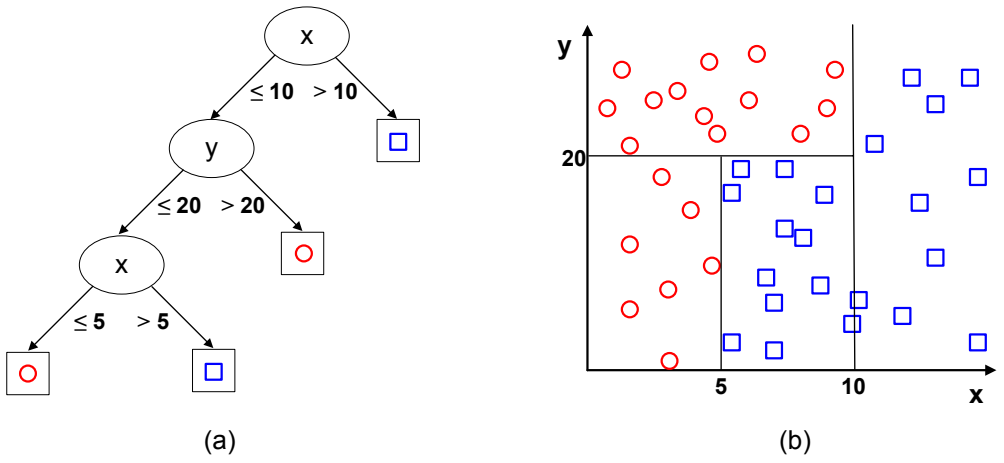

Figura 3.2: Exemplo de árvore de decisão. (a) Árvore. (b) Representação do particionamento do espaço.

3.1.1.2 Regras de Associação

As técnicas de geração de regras de associação, que geralmente são empregadas em domínios de dados categóricos, podem ser adaptadas para mineração em domínios de dados de características de baixo nível de imagens, que geralmente são compostos por dados contínuos. O objetivo dessas técnicas é a geração de regras que indiquem correlações interessantes entre características dos dados. Em [Felipe et al., 2006] o algoritmo de geração de regras de associação StarMiner [Ribeiro et al., 2005] é utilizado para selecionar as características de textura mais importantes retornadas por um extrator de momentos de Zernike, com o objetivo de melhorar a qualidade da recuperação por conteúdo de imagens de mamografias.

3.1.2 Aprendizado Não-Supervisionado

A tarefa típica que emprega o aprendizado não-supervisionado para recuperação de imagens é determinar como as imagens podem ser organizadas ou agrupadas [Liu et al., 2007]. O objetivo é criar agrupamentos de dados, minimizando a dissimilaridade entre as imagens de um agrupamento e maximizando a dissimilaridade entre agrupamentos distintos. Os algoritmos para detecção de agrupamentos podem ser divididos em duas categorias: algoritmos de particionamento, que têm como objetivo a construção de um único nível de partição que divida os dados em um número desejado de agrupamentos; e os algoritmos hierárquicos, que criam uma hierarquia de agrupamentos, formada por vários níveis de partições aninhadas do conjunto de dados. Em [Jain et al., 1999] há uma descrição detalhada desses métodos. São exemplos de algoritmos de particionamento os métodos *k-means* e *k-medoids* [Kaufman e Rousseeuw, 2005] e de algoritmos hierárquicos os métodos Single-Link [Jain e Dubes, 1988] e CURE (*Clustering Using REpresentatives*) [Guha et al., 1998]. Em seguida, as estatísticas que medem as variações existentes entre os agrupamentos podem ser utilizadas para derivar um conjunto de mapeamentos entre as características de baixo nível e um conjunto de palavras chaves, permitindo a criação de um classificador de imagens. Essa estratégia é empregada no sistema apresentado em [Jin et al., 2004]. Em [Chen et al., 2003] é apresentada a técnica denominada CLUE que tem por objetivo retornar agrupamentos de imagens considerando não somente a dissimilaridade para o elemento de consulta, mas também a dissimilaridade entre os elementos da resposta.

3.1.3 Redução de Dimensionalidade

Diversas técnicas de aprendizado supervisionado atingem resultados de excelente qualidade. Entretanto, um dos problemas dos algoritmos convencionais de aprendizado super-

visionado está relacionado à grande quantidade de amostras necessária para o treinamento, sendo que em geral, os conjuntos de amostras são estáticos durante a fase de aplicação do treinamento. Outro problema está relacionado ao tempo de processamento requerido, que normalmente torna seu uso proibitivo para grandes conjuntos de dados [Liu et al., 2007].

Técnicas de redução de dimensionalidade podem ser empregadas para diminuir a complexidade dos dados, com a conseqüente melhora do desempenho de diversas técnicas de classificação, contribuindo para a diminuição da descontinuidade semântica na recuperação de imagens. Em geral, o grande número de dimensões dos conjuntos de dados aumenta a complexidade das técnicas de manipulação e degrada o desempenho dos diversos algoritmos. Para diminuir esse efeito, as técnicas de redução de dimensionalidade têm por objetivo representar um conjunto de dados de dimensão E em outro espaço de dimensão menor que E, mantendo as características do conjunto. Os processos de redução de dimensionalidade podem ser divididos em processos de extração de atributos e processos de seleção de atributos.

Um processo de extração de atributos altera a representação de um conjunto de dados, de modo que a nova representação tenha um número menor de dimensões em relação à representação original, procurando manter as características inerentes da informação armazenada [Hair-Jr et al., 1995]. Dentre as principais técnicas para extração de atributos baseados no cálculo de distância entre objetos estão os métodos Escala Multidimensional (*Multidimensional Scaling* – MDS) [Kruskal e Wish, 1978], *MetricMap* [Wang et al., 1999] e *FastMap* [Faloutsos e Lin, 1995]. Os métodos de análise de multiresolução também podem ser utilizados para extração de atributos, dentre eles as *wavelets* de Daubechies [Daubechies, 1990], Haar (proposto em 1909 por Alfred Haar) e Gabor [Gabor, 1946]. Esses métodos podem ser utilizados tanto para a redução da resolução das imagens quanto para a redução da dimensionalidade dos vetores de características extraídos das mesmas.

Um processo de seleção de atributos escolhe um subconjunto de dimensões do conjunto de dados de acordo com uma medida de importância, sendo que não há alteração dos valores contidos nas dimensões escolhidas. Os algoritmos de aprendizado CN2 [Clark e Niblett, 1989] e C4.5 [Quinlan, 1993] são exemplos de algoritmos que podem ser utilizados para a seleção de atributos. Outras técnicas recentes têm utilizado o cálculo da dimensão fractal para a seleção de atributos significativos [Traina-Jr et al., 2000, Pagel et al., 2000, Sousa et al., 2002, Lee et al., 2006].

3.2 Técnicas de Realimentação de Relevância

Comparadas com os algoritmos de processamento estáticos discutidos na seção anterior, as técnicas de realimentação de relevância (*relevance feedback – RF*) geralmente são executadas dinamicamente com o objetivo de captar a intenção de um usuário em tempo real. Essas técnicas foram introduzidas na década de 1970 para recuperação de documentos de texto, e ultimamente seu uso tem sido considerado para a recuperação de imagens por conteúdo para a redução da descontinuidade semântica. Seu sucesso na recuperação de imagens deve-se ao fato de que uma imagem revela seu conteúdo ao usuário quase que instantaneamente, enquanto que o julgamento de documentos de texto pode levar um tempo considerável [Zhou e Huang, 2003].

A realimentação de relevância é uma estratégia de aprendizado em tempo real que adapta a resposta de um sistema de recuperação por meio da exploração da interação do usuário. É um processo que ajusta automaticamente uma consulta existente por meio da informação 'realimentada' pelo usuário sobre a relevância das respostas retornadas previamente, de modo que a consulta ajustada seja uma melhor aproximação das necessidades de informação e preferência do usuário [Doulamis e Doulamis, 2006].

Assumindo que as características de baixo nível estejam de alguma forma correlacionadas com os conceitos semânticos de alto nível, ainda há a necessidade de julgamento do usuário, uma vez que as características extraídas de imagens residem em um espaço de representação contínuo, e os conceitos semânticos são melhores descritos em subespaços discriminativos. Por exemplo, "carro" pode ser descrito por uma ou mais formas enquanto que "pôr do sol" pode ser descrito por um descritor de cores, de modo que em geral apenas um subconjunto de características é necessário para a descrição de um dado conceito. Essa correlação entre as características de baixo nível e os conceitos semânticos de alto nível geralmente não estão disponíveis *a priori*, são inerentes a cada domínio de imagens e são de difícil descoberta. Outro fator que deve ser levado em consideração é que usuários diferentes em momentos diferentes podem ter interpretações diferentes para uma mesma imagem. Desse modo, o melhor resultado que pode ser obtido é relativo à intenção de cada usuário em um dado momento, sendo extremamente importante a captura da sua individualidade [Zhou e Huang, 2003].

A Figura 3.3 apresenta o diagrama de um sistema de recuperação de imagens por conteúdo com realimentação de relevância, com destaque para o laço da realimentação. Um cenário típico de realimentação de relevância para a recuperação de imagens é composto pelos seguintes passos:

1. o sistema retorna os resultados iniciais de uma consulta baseada em um exemplo;

2. o usuário julga os resultados retornados informando um grau de relevância para um conjunto de imagens;

3. o sistema aprende com a realimentação do usuário, processa novamente a consulta, e retorna ao passo 2.

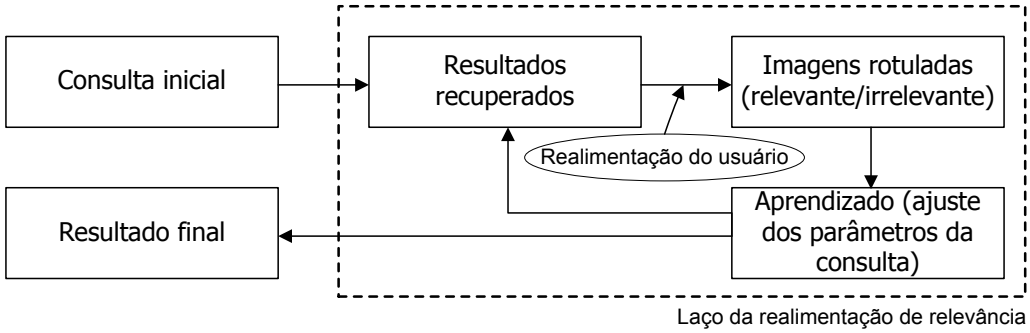

Figura 3.3: Diagrama de um sistema de recuperação de imagens por conteúdo típico com realimentação de relevância. Destaque para o laço de realimentação.

Os passos 2 e 3 podem ser repetidos até o usuário ficar satisfeito com os resultados. Para o passo 2, alguns algoritmos assumem a realimentação binária para exemplos positivos e negativos [Yin et al., 2005, Ferecatu et al., 2005, Rahmani et al., 2005]; outros assumem apenas a realimentação positiva [Chen et al., 2001]; outros assumem ainda exemplos positivos e negativos com graus de irrelevância [Zhou et al., 2006, Zhou et al., 2005]. Entre as estratégias que podem ser empregadas no passo 3 estão: a distribuição de pesos nos vetores de características de baixo nível; a movimentação do centro de consulta; e a movimentação de múltiplos centros de consulta. Essas estratégias são descritas a seguir.

3.2.1 Distribuição de Pesos

O ajuste de pesos é uma estratégia simples para aproximar os resultados das consultas das intenções do usuário. A idéia é utilizar as imagens realimentadas pelo usuário para computar o peso de cada posição do vetor de características na função de distância utilizada, tirando do usuário a responsabilidade de atribuir os pesos para as características. Em [Jing et al., 2003] é apresentada a técnica de distribuição de pesos denominada *Region Frequency versus Inverse Image Frequency* utilizando a função de distância L_1 e a realimentação apenas de exemplos positivos. Em [Doulamis e Doulamis, 2004] é utilizada a análise funcional para estimar a contribuição de cada componente do vetor de características. A Figura 3.4 apresenta uma ilustração da distribuição de pesos atribuídos às posições dos vetores de características, considerando a função de distância L_2.

Em [Rui et al., 1997] é apresentado o sistema MARS (*Multimedia Analysis and Retrieval System*) que utiliza o método do desvio padrão para analisar a importância de

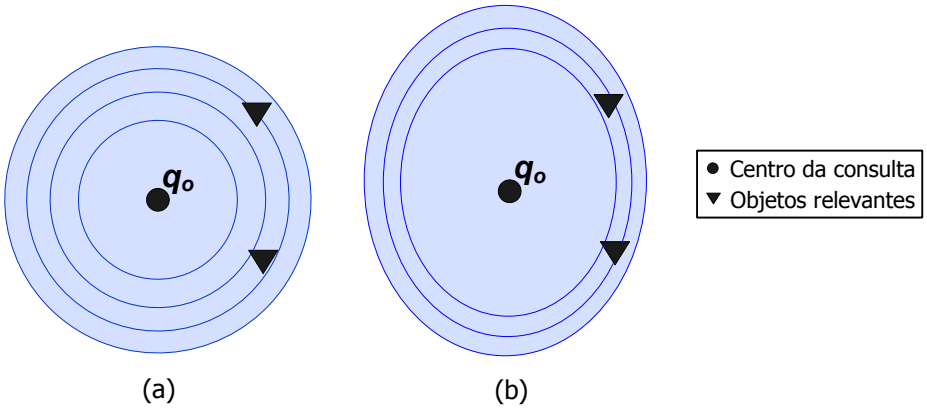

Figura 3.4: Ilustração da distribuição de pesos considerando a função de distância L_2. (a) Consulta inicial e escolha dos objetos relevantes. (b) Abrangência da função de distância alterada pelos pesos atribuídos às posições dos vetores.

cada dimensão dos vetores de características. A idéia básica desse método é associar um peso alto às dimensões que apresentarem valores similares às respectivas dimensões nas imagens do conjunto resposta rotuladas como relevantes e um peso baixo para as dimensões que apresentarem valores diferentes das respectivas dimensões nas imagens rotuladas como relevantes. Assim, o valor do peso que deve ser associado a cada dimensão do vetor de características na função de distância é determinado pelo inverso do desvio padrão dos valores da dimensão em questão, considerando todos os vetores de características das imagens do conjunto resposta rotulados como relevantes.

3.2.2 Movimentação do Centro de Consulta

A idéia básica da movimentação do centro de consulta (*Query Point Movement* – QPM) é mover o centro de uma consulta no sentido dos exemplos positivos fornecidos pelo usuário e afastar dos exemplos negativos. Em domínios multidimensionais, em geral, o resultado da movimentação do centro de consulta único é computado pela média dos vetores de características das imagens rotuladas como relevantes. A Figura 3.5 apresenta um exemplo de movimentação do centro de consulta.

A cada iteração da realimentação de relevância, o usuário informa algumas imagens como relevantes e em algumas propostas o usuário pode determinar o grau de relevância de cada uma. Em seguida, o novo centróide é calculado, levando em consideração o grau de relevância de cada imagem, e então ele é utilizado para selecionar as imagens na consulta refinada. Essa estratégia foi utilizada em [Rui et al., 1998, Ishikawa et al., 1998, Liu et al., 2006a, Doulamis e Doulamis, 2006, Liu et al., 2006b, Shen et al., 2009].

Considerando a realimentação composta por exemplos positivos e negativos, a fórmula de Rocchio [Rocchio, 1971] pode ser empregada para a movimentação do centro de

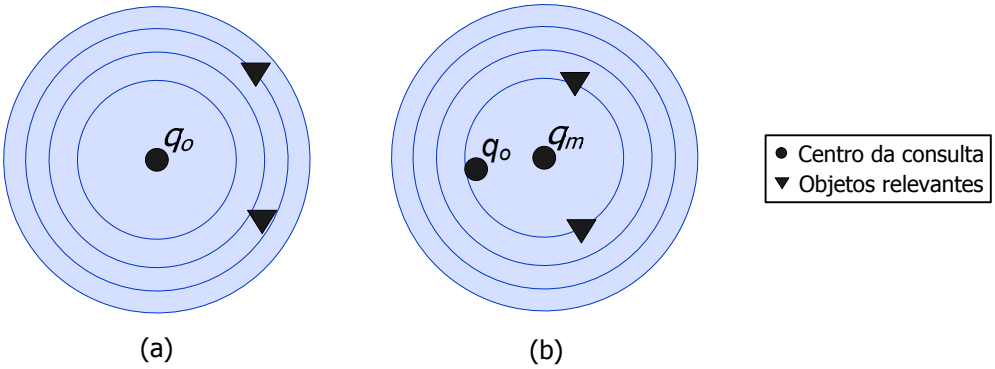

Figura 3.5: Ilustração da movimentação do centro de consulta. (a) Consulta inicial e escolha dos objetos relevantes. (b) Centro de consulta q_o movimentado para posição q_m.

consulta. A fórmula de Rocchio é definida pela Equação 3.1, sendo que: Q é o vetor de características original e Q' é o vetor de características movimentado; D'_R e D'_N são respectivamente os exemplos positivos e negativos realimentados; $N_{R'}$ e $N_{N'}$ correspondem respectivamente à quantidade de exemplos em D'_R e D'_N; e α, β e γ são constantes selecionadas, que determinam o peso do elemento de consulta, o peso da realimentação positiva e o peso da realimentação negativa, respectivamente na movimentação do centro de consulta.

$$Q' = \alpha Q + \beta \left(\frac{1}{N_{R'}} \sum_{i \in D'_R} D_i \right) - \gamma \left(\frac{1}{N_{N'}} \sum_{i \in D'_N} D_i \right) \tag{3.1}$$

A fórmula de Rocchio maximiza as diferenças entre o centro de consulta Q e as imagens selecionadas como irrelevantes D'_N e minimiza as diferenças entre o centro de consulta Q e as imagens selecionadas como relevantes D'_R. Desse modo, todos os elementos relevantes e irrelevantes são empregados para mover o centro de consulta Q, adicionando as diferenças normalizadas de D'_R e subtraindo as diferenças normalizadas de D'_N, resultando em um novo elemento Q'. É importante notar que a movimentação do elemento de consulta cria um novo vetor de características que pode não corresponder a uma imagem do conjunto de dados.

Em [Traina et al., 2006] é apresentada uma técnica de movimentação do centro de consulta que analisa cada atributo dos vetores separadamente, determinando valores mínimos e máximos de movimentação do atributo baseado nos valores do atributo das imagens realimentadas como relevantes. O objetivo é limitar a movimentação resultante das imagens realimentadas como irrelevantes (realimentação negativa), não permitindo o deslocamento do centro para regiões distantes do espaço de busca.

Em [Liu et al., 2006a, Liu et al., 2006b] são apresentadas técnicas de movimentação do centro de consulta que permitem a convergência em um menor número de iterações

do laço de realimentação de relevância. Elas são baseadas em amostragem aleatória e na análise da vizinhança local por meio da construção de diagramas de Voronoi para a realização da nova consulta aos vizinhos mais próximos.

3.2.3 Movimentação de Múltiplos Centros de Consulta

As técnicas baseadas na movimentação de múltiplos centros de consulta podem ser separadas em três técnicas principais, denominadas: expansão da consulta; abordagem *Qcluster*; e abordagem *Top-k*.

3.2.3.1 Expansão da Consulta

Na técnica de expansão da consulta (*Query Expansion* – QEX), as imagens informadas como relevantes são organizadas em agrupamentos, de modo que cada agrupamento é representado por uma imagem mais próxima do centro do agrupamento. Esses novos centros são tratados como representantes, e são usados em uma consulta de múltiplos centros, sendo que o número de imagens que cada centro representa é utilizado como peso na nova consulta. Assim, a distância de um objeto em uma consulta de múltiplos centros corresponde a uma combinação ponderada das distâncias individuais entre o objeto e os representantes da consulta. Essa estratégia foi utilizada em [Hua et al., 2006] e [Porkaew et al., 1999]. A Figura 3.6 apresenta um exemplo de expansão da consulta com dois novos centros escolhidos por meio das imagens rotuladas como relevantes.

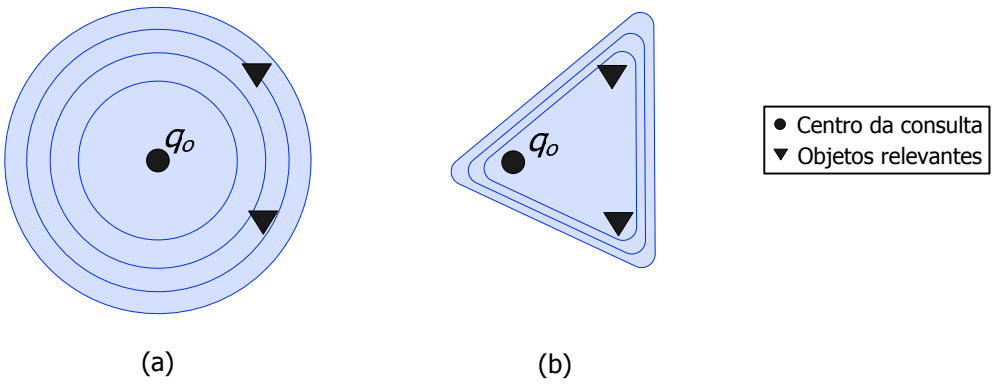

Figura 3.6: Ilustração da movimentação de múltiplos centros de consulta, abordagem de expansão da consulta. (a) Consulta inicial e escolha dos objetos relevantes. (b) Movimentação de formato convexo de múltiplos centros de consulta.

3.2.3.2 Abordagem *Qcluster*

A abordagem de expansão da consulta assume que as imagens relevantes são mapeadas para elementos próximos entre si, de acordo com a medida de dissimilaridade, e um único

contorno é traçado para cobrir todas as imagens da consulta. Entretanto, é possível que as imagens realimentadas como relevantes formem agrupamentos disjuntos. Logo, o contorno traçado pode conter regiões de imagens irrelevantes no espaço das características. Para superar esse problema foi proposto em [Kim e Chung, 2003] um método de agrupamento adaptativo que consiste em dois processos: classificação e combinação de agrupamentos. O processo de classificação atribui cada imagem realimentada como relevante em um dos agrupamentos correntes ou em um novo agrupamento. Em seguida, o processo de combinação de agrupamentos reduz o número de agrupamentos por meio da combinação de certos agrupamentos de conteúdo similar, diminuindo também o número de imagens na consulta em cada iteração. Finalmente as imagens representativas dos agrupamentos são usadas na consulta de múltiplos centros. A Figura 3.7 apresenta a idéia básica da abordagem *Qcluster*.

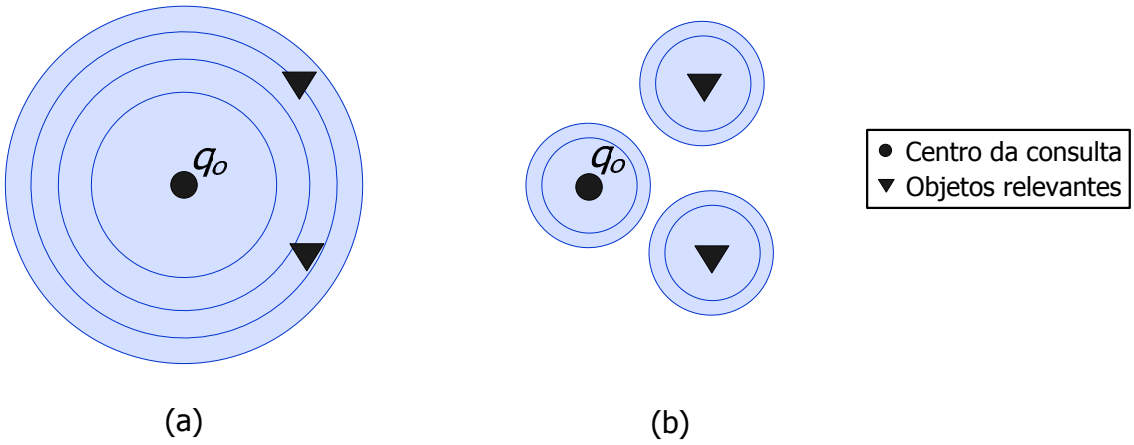

Figura 3.7: Ilustração da movimentação de múltiplos centros de consulta, abordagem *Qcluster*. (a) Consulta inicial e escolha dos objetos relevantes. (b) Movimentação de formato concavo de múltiplos centros de consulta.

3.2.3.3 Abordagem *Top-k*

Na abordagem *Top-k* [French et al., 2004, Jin e French, 2005], ao invés de agrupar as imagens marcadas como relevantes antes de executar a consulta, as consultas aos múltiplos centros são executadas individualmente e os resultados são combinados para melhorar o desempenho da recuperação. A Figura 3.8 apresenta a diferença entre as abordagens citadas anteriormente (a) e a abordagem *Top-k* (b). Após a execução das consultas *Top-k* considerando as imagens marcadas como relevantes como centros, as imagens são selecionadas entre todas as imagens retornadas pela dissimilaridade entre cada imagem retornada e a imagem usada para a consulta *Top-k*.

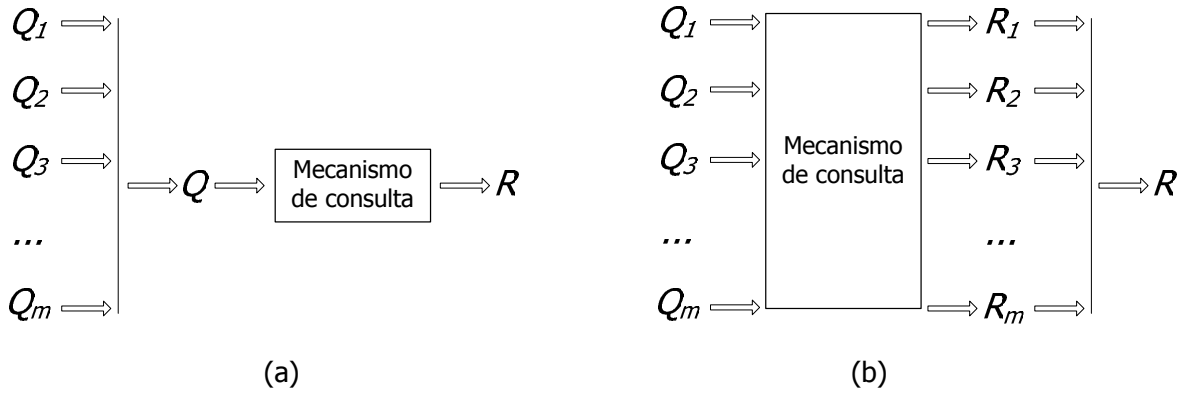

Figura 3.8: Representação da abordagem *Top-k*. (a) Combinação das consultas de múltiplos centros. (b) Combinação dos resultados das consultas.

3.2.4 Semântica e Avaliação das Técnicas de Realimentação de Relevância

Em [Doulamis e Doulamis, 2006] várias técnicas de realimentação de relevância são avaliadas. As técnicas melhoram a semântica dos resultados das consultas nos sistemas de recuperação de imagens por conteúdo por meio da adaptação das respostas com a informação das necessidades e preferências do usuário. Em geral, a precisão desses sistemas para uma dada revocação aumenta de acordo com as iterações da realimentação. Entretanto, experimentos comprovam que a taxa de aumento da precisão diminui a partir de uma determinada quantidade de iterações de realimentação executadas [Traina et al., 2006].

As técnicas de realimentação de relevância baseadas na distribuição de pesos apresentam consistência no caso do usuário ser consistente nas suas realimentações, logo, a convergência é alcançada sob a condição de que o usuário envie informações de relevância consistentes ao sistema. Em geral, essas técnicas requerem a reorganização das estruturas de indexação envolvidas, uma vez que modificam a métrica utilizada.

As técnicas de realimentação de relevância baseadas na movimentação do centro de consulta permitem a integração eficiente com estruturas de indexação baseadas em distâncias, evitando o acesso seqüencial ao conjunto de dados em questão. As técnicas de movimentação de múltiplos centros resultam em consultas de maior qualidade semântica em relação à movimentação de um único centro de consulta. A movimentação de múltiplos centros pode ser realizada como uma única ação (por exemplo, abordagem de expansão da consulta) ou como a combinação do resultado de múltiplas consultas (por exemplo, a abordagem *Top-k*). Os experimentos apresentados em [Doulamis e Doulamis, 2006] indicam que a combinação de resultados levam a um melhor desempenho do que a movimentação realizada como uma única ação, uma vez que imagens rotuladas como relevantes podem estar localizadas em agrupamentos diferentes no espaço das características.

3.3 Tipos de Relevância

O propósito dos sistemas de informação é a recuperação de itens relevantes. A noção de relevância é fundamental para o processo de recuperação de informação, e em geral, um usuário tem uma idéia clara do que é relevante, reflexo do objetivo da sua busca.

Uma das definições mais simples e mais usadas de relevância é a relação binária entre o elemento de consulta e um elemento do conjunto de dados, que define como relevante ou irrelevante o elemento do conjunto. Entretanto, diversos autores propõem o uso de uma noção de graduação da relevância, que pode ser tanto baseada em categorias discretas ou intervalos entre valores reais. Nesses casos, a concordância entre usuários diferentes a respeito da relevância de um elemento tende a ser mais baixa que a concordância da relevância binária. Além disso, há estudos que indicam que os usuários preferem usar os extremos de uma dada escala [Lavrenko, 2009]. Ainda assim, a definição de múltiplos níveis de relevância permite a criação de modelos interessantes, que muitas vezes permitem mapear o desejo dos usuários com maior precisão.

3.4 Otimização de Técnicas de Realimentação de Relevância

A maioria das propostas de técnicas de realimentação de relevância em consultas por conteúdo de imagens aborda apenas a qualidade semântica dos métodos. Entretanto, alguns trabalhos apresentam métodos de otimização para as técnicas propostas, para torná-las escaláveis para grandes conjuntos de dados. Um dos trabalhos pioneiros na utilização de realimentação de relevância em CBIR propõe o sistema *MindReader* [Ishikawa et al., 1998], no qual os elementos realimentados alteram a função de distância euclidiana gerando regiões elípticas que podem ser otimizadas por métodos de acesso espaciais. Em [Wu et al., 2000], uma consulta de múltiplos centros baseada nos elementos escolhidos durante a realimentação de relevância resulta na união de consultas por abrangência de único centro que são otimizadas pelo método de acesso métrico *M-tree* [Ciaccia et al., 1997] para posterior aplicação de filtro. Em [Wu e Manjunath, 2001] a realimentação de relevância é empregada na computação dos pesos aplicados às dimensões de uma função de distância, e um método de acesso permite que uma aproximação da consulta seja realizada baseada na função de distância resultante. Em [Shen et al., 2009] é proposto um método que realiza uma predição e seleção dos potenciais candidatos a resposta nas próximas iterações do ciclo de realimentação, reduzindo o número de leituras à memória secundária, por meio da computação da sobreposição entre duas iterações consecutivas utilizando um método de regressão linear.

3.5 Diversidade em Consultas por Conteúdo de Imagens

As consultas aos k-vizinhos mais próximos retornam os k elementos menos dissimilares ao elemento de consulta de acordo com uma função de distância δ. Essa abordagem leva em consideração a relevância individual de cada elemento computado pela função de distância aplicada aos seus atributos com relação ao elemento de consulta. Entretanto, ela não considera o relacionamento desses elementos entre si.

Uma forma de aumentar a relevância nas consultas baseadas no conteúdo de imagens é prover diversidade nos resultados. Em geral, a falta de diversidade em uma consulta às imagens mais próximas de uma imagem de consulta tende a prejudicar a busca, uma vez que o usuário precisa analisar um grande número de imagens para encontrar imagens relevantes. A diversidade em consultas aos k-vizinhos mais próximos vem sendo estudada ultimamente no contexto de recuperação de informação, como no caso de máquinas de busca da *web* e de aplicações de comércio eletrônico [Vee et al., 2008, Xin et al., 2006], que resultaram em algoritmos específicos para recuperação baseada em texto.

A Figura 3.9 apresenta um exemplo de aplicação de consulta *top-k* em uma máquina de busca da *web*. Nesse exemplo, foi realizada uma consulta à palavra chave *Faloutsos* que corresponde ao sobrenome de três pesquisadores em computação. A diversidade apresentada na primeira página de resultados permite ao usuário refinar a sua busca, caso o mesmo esteja pesquisando a respeito de um determinado indivíduo. No exemplo, a consulta retornou como mais relevantes os *links* para as páginas pessoais dos três pesquisadores com esse sobrenome, os Professores Christos, Petros e Michalis. É interessante notar que, sem o tratamento da diversidade, a consulta poderia ter retornado na primeira página de resultados os *links* para documentos de apenas um dos autores, por exemplo, o autor com maior número de publicações.

A Figura 3.10 refere-se à busca de imagens da *web* baseado em elementos de texto localizados próximos às imagens, como legendas, que também procurou diversificar os resultados. Nesse caso, a integração com técnicas de busca por conteúdo de imagens poderia aumentar a semântica dos resultados, permitindo, por exemplo, retornar apenas as imagens que correspondem a formas de pessoas.

A computação da diversidade em uma consulta aos vizinhos mais próximos pode ser dada de várias maneiras. Em [Jain et al., 2004], a diversidade é dada por função booleana combinada com a agregação das distâncias para o elemento de consulta. Agrawal et al. [Agrawal et al., 2009] avaliam a diversidade de documentos que podem pertencer a mais de uma categoria de acordo com uma taxonomia. Em [van Leuken et al., 2009] as imagens retornadas em uma consulta são submetidas a uma técnica de detecção de

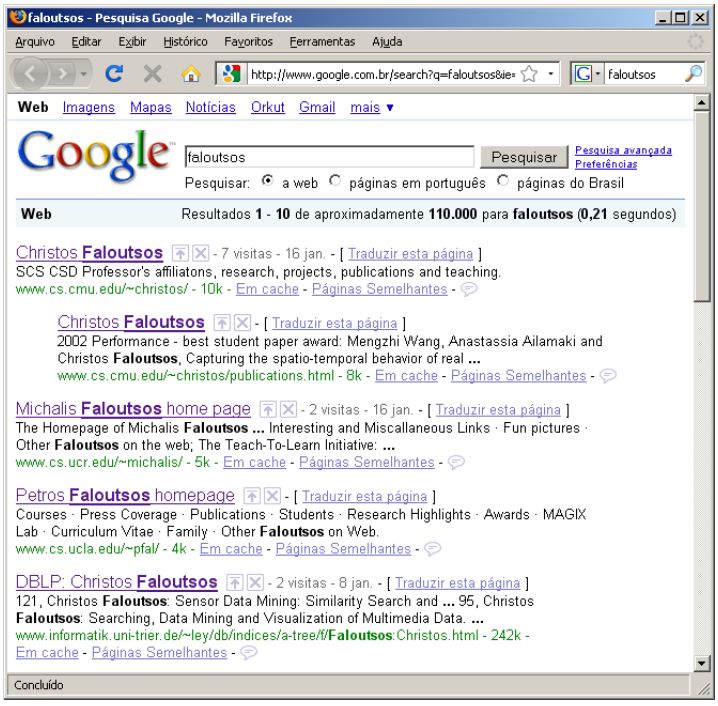

Figura 3.9: Diversidade em uma consulta em uma máquina de busca da *web*. A consulta à palavra-chave "Faloutsos" retornou, entre os primeiros registros, os *links* para as páginas pessoais dos três principais pesquisadores com esse sobrenome (Professores Christos, Michalis e Petros). Consulta realizada em 18/05/2009.

agrupamentos e o elemento central de cada agrupamento é adicionado ao conjunto de elementos diversos resultante. Em [Vee et al., 2008] a diversidade é dada por valores distintos de uma sequência de atributos de uma relação.

A diversidade no contexto de recuperação de imagens baseada em conteúdo foi pouco explorada, especialmente no modelo em que apenas distâncias entre imagens podem ser computadas. Isso levou ao desenvolvimento do trabalho apresentado no Capítulo 5 desta tese, que aborda o tratamento da diversidade em consultas aos k-vizinhos mais próximos, de modo que a diversidade é dada pela computação de distâncias entre os elementos da resposta.

3.5.1 Problema da Diversidade Máxima

Um problema relacionado com o tratamento de diversidade em consultas aos vizinhos mais próximos é denominado problema da diversidade máxima. O problema da diversidade máxima (*Maximum Diversity Problem – MDP*) consiste em selecionar um conjunto ótimo de k elementos diversos de um conjunto de dados. O processo de seleção dos elementos baseia-se na computação de distâncias entre os pares de elementos, de modo que o objetivo é encontrar uma solução que corresponda a um conjunto de k elementos que apresente a máxima diversidade possível.

Figura 3.10: Recuperação de imagens baseado em elementos de texto em uma máquina de busca da *web*. Consulta à palavra-chave "Faloutsos" resultou em imagens com diversidade entre os atributos de texto relacionados com as imagens. Consulta realizada em 18/05/2009.

A busca por tal solução pertence a classe de problemas NP-difíceis [Kuo et al., 2007, Ghosh, 1996]. Assim, muitos trabalhos desenvolveram heurísticas para obter soluções aproximadas do problema, como as baseadas na meta-heurística GRASP (*Greedy Randomized Adaptive Search Procedure*) [Feo e Resende, 1995, Laguna e Marti, 1999, Resende e Ribeiro, 2003, Silva et al., 2007, Resende, 2009]. Uma visão geral desses métodos pode ser encontrada em [Resende, 2009].

3.5.1.1 GRASP

A meta-heurística GRASP é um método de otimização das funções objetivo baseado em busca aleatória [Feo e Resende, 1995, Resende, 2009]. É um processo iterativo, no qual cada iteração consiste em duas fases principais, denominadas construção e busca local. O objetivo da fase de construção é selecionar uma solução razoável cuja vizinhança deverá ser explorada na fase de busca local. As duas fases são repetidas até que um critério de parada seja alcançado, e então a melhor solução encontrada é retornada. O Algoritmo 1 apresenta os passos principais do método GRASP que emprega um número máximo de iterações como critério de parada.

A estratégia empregada na construção da solução inicial consiste em selecionar alea-

Algoritmo 1 Meta-heurística GRASP.

Entrada: *maxIterações*

```
1: solução ← ∅
2: melhorSolução ← ∅
3: para i ← 0 to maxIterações faça
4:    solução ← FaseDeConstrução()
5:    solução ← FaseDeBuscaLocal(solução)
6:    se solução > melhorSolução então
7:       melhorSolução ← solução
8:    fim se
9: fim para
10: retorne melhorSolução
```

toriamente um elemento de uma lista restrita de candidatos (*Restricted Candidate List – RCL*) de tamanho fixo a cada iteração da fase de construção. A computação da RCL é guiada por uma lista de candidatos, que ordena os elementos candidatos em ordem descendente de acordo com uma medida de benefício calculada por uma função de avaliação gulosa.

Com a solução retornada pela fase de construção servindo como ponto de partida, a busca local pode progressivamente melhorá-la com a aplicação de uma série de modificações locais na vizinhança da solução corrente. A fase de busca local termina quando encontra um ótimo local.

O algoritmo GRASP básico é considerado *memoryless*, ou seja, cada iteração é independente e não é influenciada pelas soluções encontradas previamente. Várias estratégias foram propostas para lidar com essa questão, que consideram as soluções anteriores como modo de melhorar a fase de construção [Fleurent e Glover, 1999, Prais e Ribeiro, 2000] e a fase de busca local [Laguna e Marti, 1999]. Dentre as estratégias, destaca-se o método *path relinking*.

3.5.1.2 Path Relinking

A idéia básica dessa estratégia consiste em buscar por melhores soluções que estiverem no caminho que conecta duas soluções conhecidas [Laguna e Marti, 1999, Resende e Ribeiro, 2005]. Segundo [Resende e Ribeiro, 2003], a estratégia *path-relinking* é efetiva quando é empregada como uma estratégia de intensificação de cada iteração de um algoritmo GRASP. Nesse caso, ela é aplicada em pares de soluções que são compostas por uma solução local ótima obtida após uma busca local de uma iteração do algoritmo GRASP e uma solução escolhida aleatoriamente entre um conjunto de soluções denominadas soluções elite, encontradas durante as buscas locais anteriores. Uma solução ótima local é incluída no conjunto de soluções elite se for suficientemente diferente das outras

soluções elite e também for melhor que a pior dessas soluções.

A exploração da trajetória que conecta um par de soluções é precedida pela computação de um conjunto de movimentações que devem ser realizadas em uma das soluções para transformar-se na outra solução. Ao final do processo, a melhor solução encontrada ao longo da trajetória é também considerada para ser incluída no conjunto de soluções elite e também para atualizar a melhor solução conhecida [Ribeiro et al., 2002].

3.6 Avaliação de Desempenho de Consultas

A análise de algoritmos de otimização é uma tarefa árdua. Para uma dada instância de um problema, um algoritmo A pode encontrar uma solução melhor que a solução de um algoritmo B e para outra instância do problema, B pode encontrar uma solução melhor. Assim, a execução desses algoritmos para um grande conjunto de instâncias é uma forma de comparar algoritmos nesses experimentos. Grande parte dos testes de desempenho (*benchmarks*) apresentam tabelas com o desempenho dos algoritmos para um conjunto de instâncias de um problema, e a análise dessas tabelas pode ser uma fonte de discordância. Nesse contexto, Dolan e Moré [Dolan e Moré, 2002] desenvolveram o método denominado perfis de desempenho (*performance profiles*) para comparar o desempenho de um conjunto de algoritmos em um conjunto de problemas.

Considere-se que há um conjunto $\mathcal{S}$ com n_s algoritmos e um conjunto $\mathcal{P}$ com n_p problemas. Para cada problema p e algoritmo s, tem-se:

$$t_{p,s} = \text{resultado do problema } p \text{ pelo algoritmo } s$$

O tempo de execução, o número de cálculos de distância ou outra medida de avaliação pode ser considerado como resultado de um algoritmo para um dado problema. A base de comparação é o resultado do melhor desempenho do problema p entre todos os n_s algoritmos sob análise. A razão de desempenho $r_{p,s}$ é dada pela Equação 3.2:

$$r_{p,s} = \frac{t_{p,s}}{min\{t_{p,s} : s \in \mathcal{S}\}} \tag{3.2}$$

O desempenho de um algoritmo s para um dado problema pode ser interessante, mas a avaliação global do desempenho de um algoritmo pode determinar a escolha do mesmo. A Equação 3.3 define $\rho_s(\tau)$ como a probabilidade de um algoritmo $s \in \mathcal{S}$ ter a razão de desempenho $r_{p,s}$ até um fator $\tau \in \mathbb{R}$ da melhor razão possível. A função ρ_s é a função de distribuição cumulativa da razão de desempenho.

$$\rho_s(\tau) = \frac{1}{n_p} |\{p \in \mathcal{P} : r_{p,s} \leq \tau\}| \tag{3.3}$$

O termo perfil de desempenho refere-se a função de distribuição de uma métrica de avaliação. Um perfil de desempenho é uma função constante por partes, não-decrescente e contínua à direita de cada ponto de descontinuidade, que resulta na probabilidade de um algoritmo obter melhor desempenho que os outros algoritmos, correspondendo a uma função de distribuição acumulada para a razão de desempenho. Se o conjunto de problemas $\mathcal{P}$ representa suficientemente os problemas que podem ocorrer em uma aplicação, então os algoritmos com maior probabilidade $\rho_s(\tau)$ devem ser escolhidos. O valor $\rho_s(1)$ é a probabilidade de que um algoritmo terá melhor desempenho sobre os outros algoritmos.

Os gráficos resultantes contém uma curva para cada algoritmo sob análise, no qual quanto mais próximo do topo do gráfico, melhor é o desempenho do algoritmo. A Figura 3.11 apresenta um exemplo de perfis de desempenho no qual é possível verificar que o algoritmo 1 obteve melhor desempenho que o algoritmo 2 que por sua vez obteve melhor desempenho que o algoritmo 3.

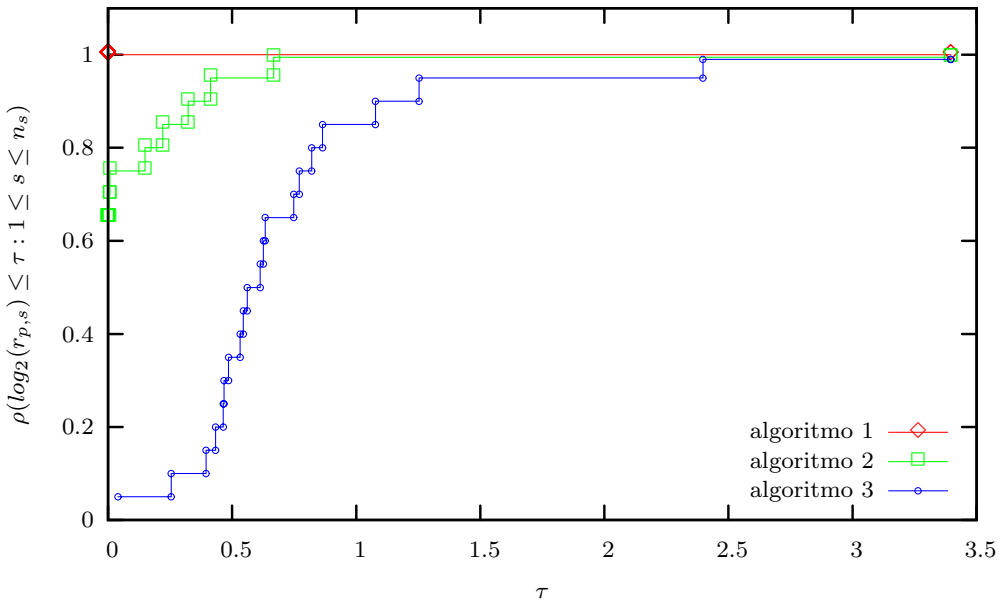

Figura 3.11: Exemplo de gráfico de perfis de desempenho.

Para permitir a comparação dos métodos, também pode ser computada a medida *gap* entre cada método proposto e um método de referência. A medida *gap* é computada conforme a Equação 3.4:

$$gap = \frac{\text{pontuação}_r - \text{pontuação}_c}{\text{pontuação}_r} \cdot 100\% \tag{3.4}$$

sendo que pontuação_r é a medida da função de pontuação do método de referência e

pontuação$_c$ é a pontuação do método em comparação.

O *gap* é o incremento de qualidade com relação à estratégia de referência, assim, quanto maior o valor melhor o resultado. Essa medida de avaliação vem sendo utilizada em trabalhos relacionados à otimização de problemas com complexidade de tempo polinomial ou NP que recorrem à soluções heurísticas, como em [Nascimento e Toledo, 2008].

3.7 Considerações Finais

Este capítulo apresentou as abordagens para o tratamento da descontinuidade semântica entre as características de baixo nível extraídas automaticamente de imagens e a subjetividade da percepção humana, no qual foram apresentados os conceitos de aprendizado de máquina, de realimentação de relevância e de diversidade. Ainda não existem estudos conclusivos que definam como incorporar o tratamento da descontinuidade semântica para recuperação de imagens por conteúdo em um SGBD. Entretanto, a eficiência e otimização desses métodos deve ser considerada. No próximo capítulo, são apresentados consultas que podem ser empregadas em métodos de realimentação de relevância, bem como a otimização das mesmas baseadas em métodos de acesso métrico.

Capítulo

4

Consultas por Similaridade Agregada

Dentre as principais técnicas de realimentação de relevância estão as técnicas baseadas em múltiplos centros de consulta [Doulamis e Doulamis, 2006], sendo que encontrar o conjunto de elementos que minimizam a agregação das distâncias para o conjunto de centros de uma consulta é um modo de se executar uma consulta de múltiplos centros, como na técnica Falcon descrita nas Seções 2.3.3 e 2.4.3 desta tese. Nessas técnicas, a interação com o usuário tem como conseqüência a rotulação de um grupo de imagens com relação à sua relevância no resultado da consulta, que são empregadas como entrada para uma nova consulta, em geral composta por mais de um centro de consulta e com pesos indicando a relevância das imagens.

Em geral as consultas por similaridade são consideradas como imersas em espaços métricos, que dispõem da propriedade de desigualdade triangular que permite a otimização dessas consultas. Nesse contexto, a definição das consultas por similaridade agregada em espaços métricos cria a possibilidade de otimização dessas consultas. Assim, um resultado dessa tese foi a definição das consultas por similaridade agregada como uma generalização das consultas por similaridade baseadas em um único centro, considerando tanto as consultas por abrangência quanto as consultas aos vizinhos mais próximos em espaços métricos [Razente et al., 2008b], bem como a exploração de suas propriedades e algoritmos para execução eficiente em métodos de acesso métrico [Razente et al., 2008a].

Este capítulo visa mostrar como as consultas por similaridade agregada podem ser usadas como um mecanismo poderoso para prover realimentação de relevância em um sistema de recuperação de imagens por conteúdo. Os símbolos empregados neste capítulo são apresentados na Tabela 4.1. Ele está estruturado da seguinte forma. A Seção 4.1

apresenta a definição das consultas por similaridade considerando múltiplos centros de consulta. A Seção 4.2 apresenta a técnica de otimização desenvolvida para as consultas por similaridade agregada. A Seção 4.3 apresenta experimentos realizados em um sistema de recuperação de imagens por conteúdo e a Seção 4.4 trata das considerações finais do capítulo.

Tabela 4.1: Tabela de Símbolos

Símbolo	**Definição**
$\mathbb{S}, \mathbb{S}_j$	Domínio de dados métrico
$\mathcal{M}$	Espaço métrico
R, S	Conjuntos de dados do domínio, $R \subset \mathbb{S}, S \subset \mathbb{S}$
$\delta()$	Função de distância ou função de dissimilaridade, $\delta : \mathbb{S} \times \mathbb{S} \rightarrow \mathbb{R}^+$
$d_g()$	Função de similaridade agregada, $d_g : P(\mathbb{S}_j) \times \mathbb{S}_j \rightarrow \mathbb{R}^+$
$P(\mathbb{S})$	Conjunto potência de $\mathbb{S}$
$\wp$	Predicado de similaridade agregada
ℓ	Limite de uma consulta por similaridade
k	Número máximo de elementos a ser retornado em uma consulta
ξ	Raio agregado de consulta
ξ_m	Raio agregado mínimo
g	Fator de agregação, $g \in \mathbb{R}^*$
Q	Conjunto de centros de uma consulta por similaridade agregada
$\|Q\|$	Cardinalidade de Q
s_q	Centro de consulta, $s_q \in Q$
s_i, s_j	Elementos do conjunto de dados
s_t	Elemento central ou representante de um nodo de uma Slim-tree
r_t	Raio de cobertura de um nodo de uma Slim-tree
a, b, c, d, e, f	Distâncias entre pares de elementos conhecidos
v, w, x, y	Distâncias nas quais um dos elementos é desconhecido
t, u	Variáveis relacionadas ao raio agregado de consulta
h_1, h_2	Elementos desconhecido

4.1 Consultas por Similaridade Agregada

Definição 1. Similaridade agregada. Seguindo as definições dos espaços métricos, uma função de dissimilaridade $\delta()$ pode ser qualquer função que compara um par de elementos $s_i, s_j \in \mathbb{S}$ que atenda às propriedades de simetria, não-negatividade, identidade e desigualdade triangular. Entretanto, como o conjunto de centros de consulta Q pode ter mais de um elemento, a distância $\delta(s_i, s_q)$ entre cada centro de consulta $s_q \in Q$ e um elemento $s_i \in S$ deve ser avaliada para calcular a função de dissimilaridade agregada $d_g : P(\mathbb{S}) \times \mathbb{S} \rightarrow R^+$ entre s_i e o conjunto de centros de consulta $Q \subset P(\mathbb{S})$, sendo que $P(\mathbb{S})$ representa o conjunto potência de $\mathbb{S}$. O predicado de similaridade agregada usa os valores resultantes da agregação das distâncias para ordenar os elementos em S com respeito à Q.

Dado um conjunto de elementos $S \subset \mathbb{S}$, no qual $\mathbb{S}$ é um domínio de elementos de um espaço métrico $\mathcal{M} = \langle \mathbb{S}, \delta() \rangle$, um conjunto de centros de consulta $Q \subset \mathbb{S}$ e uma função de dissimilaridade agregada $d_g()$ que calcula a similaridade agregada de cada elemento $s_i \in S$ baseado na sua similaridade medida pela função de distância $\delta()$ para cada elemento de consulta $s_q \in Q$, então um predicado de similaridade agregada $\wp(\langle d_g(), Q, \ell \rangle) : S$ recupera todo elemento $s_i \in S$ cuja similaridade agregada computada por $d_g()$ não exceda um dado limite ℓ.

Há basicamente dois tipos de predicados por similaridade: os que limitam a resposta baseados em um dado limiar de distância ξ e os que limitam a resposta baseados no número k de elementos na resposta. Se uma função de agregação apropriada $d_g()$ for empregada, os predicados de similaridade conhecidos como abrangência (*range*) e vizinhos mais próximos (*nearest neighbors*) podem ser definidos como casos especiais de predicados de similaridade agregada $\wp$, de modo que o conjunto de centros de consulta tenha apenas um elemento $Q = \{s_q\}$ e o limite ℓ seja ou um raio de abrangência ou um número k de vizinhos mais próximos, respectivamente. Nesta tese, as consultas por similaridade agregada são definidas como segue.

Definição 2. Consulta por abrangência agregada (*aggregate range query* – *ARq*). Dado um raio agregado máximo ξ, uma função de agregação de similaridade $d_g()$ e um conjunto de centros de consulta Q, a consulta ARq recupera todo elemento $s_i \in S$, tal que $d_g(s_i, Q) \leq \xi$. Formalmente:

$$ARq(Q, \xi) = \{s_i \in S | d_g(s_i, Q) \leq \xi\}$$

Definição 3. Consulta aos k-vizinhos mais próximos agregados (*aggregate k-nearest neighbor query* – *kANNq*). Dado um número inteiro $k \geq 1$, a consulta $kANNq$ recupera os k elementos que resultam nos menores valores para a função de dissimilaridade agregada $d_g()$ dos centros de consulta Q, ordenados pela função $d_g()$. Formalmente:

$$ANNq(Q, k) = \{R \subseteq S, |R| = k \wedge \forall s_i \in R, s_j \in S - R : d_g(Q, s_i) \leq d_g(Q, s_j)\}$$

Existem muitas alternativas para definir a função de dissimilaridade agregada. Nesta tese, analisamos a função definida na Equação 4.1 como um exemplo de função de dissimilaridade agregada $d_g()$:

$$d_g(Q, s_i) = \sqrt[g]{\sum_{s_q \in Q} [\ \delta(s_q, s_i)^g \cdot w_q\ ]} \tag{4.1}$$

na qual $\delta()$ é uma função de dissimilaridade, Q é um conjunto de centros de consulta, $s_i \in S$ é um elemento do conjunto de dados S, w_q é o peso correspondente ao elemento s_q

e $g \in \mathbb{R}^*$ é um valor real diferente de zero denominado fator de agregação. Ao considerar a Equação 4.1, as consultas por abrangência agregada e as consultas aos k-vizinhos mais próximos agregados executadas com apenas um centro de consulta correspondem às consultas por similaridade tradicionais denominadas consultas por abrangência e consultas aos k-vizinhos mais próximos, respectivamente.

A Figura 4.1 apresenta exemplos de consultas ao primeiro vizinho mais próximo agregado (1-ANNq) sobre o conjunto de coordenadas geográficas das cidades brasileiras, de modo que o conjunto de centros de consulta Q = {Campo Grande-MS, São Paulo-SP, Fortaleza-CE} calculando-se as distâncias com a função *spherical law of cosines* e a função de dissimilaridade agregada definida na Equação 4.1. A similaridade foi agregada em três formas distintas, sendo que: na Figura 4.1-a a minimização da distância máxima (usando o fator de agregação $g = \infty$) retornou a cidade de Ponte Alta do Bom Jesus-TO; na Figura 4.1-b a minimização da distância média quadrática (usando o fator de agregação $g = 2$) retornou a cidade de Cabeceiras-GO; e na Figura 4.1-c a minimização da soma das distâncias (usando o fator de agregação $g = 1$) retornou a cidade de Guaíra-SP.

(a) (b) (c)

Figura 4.1: Exemplos de consultas ao primeiro vizinho mais próximo agregado. (a) Minimização da distância máxima. (b) Minimização da distância média quadrática. (c) Minimização da soma das distâncias.

A Figura 4.2 apresenta o efeito do fator de agregação g em um espaço de duas dimensões regido pela distância euclidiana, Manhattan ou Chebychev, considerando $Q = \{q_1, q_2, q_3, q_4\}$. Cada curva representa as posições geométricas onde a Equação 4.1 resulta no valor em relação à Q, correspondendo a isolinhas no espaço de duas dimensões correspondente a cada função de distância, ou isosuperfícies em um espaço métrico genérico. É importante notar que para $g < 1$ podem ser geradas regiões disjuntas.

A minimização da Equação 4.1 inclui algumas funções de interesse especial. Por exemplo, $g = 2$ define a minimização da soma das distâncias agregadas médias quadráticas e $g = 1$ define a minimização da soma das distâncias agregadas. A minimização da dissimilaridade agregada máxima ($g = \infty$) e a minimização da dissimilaridade agregada mínima ($g = -\infty$) são casos especiais da Equação 4.1 apresentadas respectivamente nas Equações

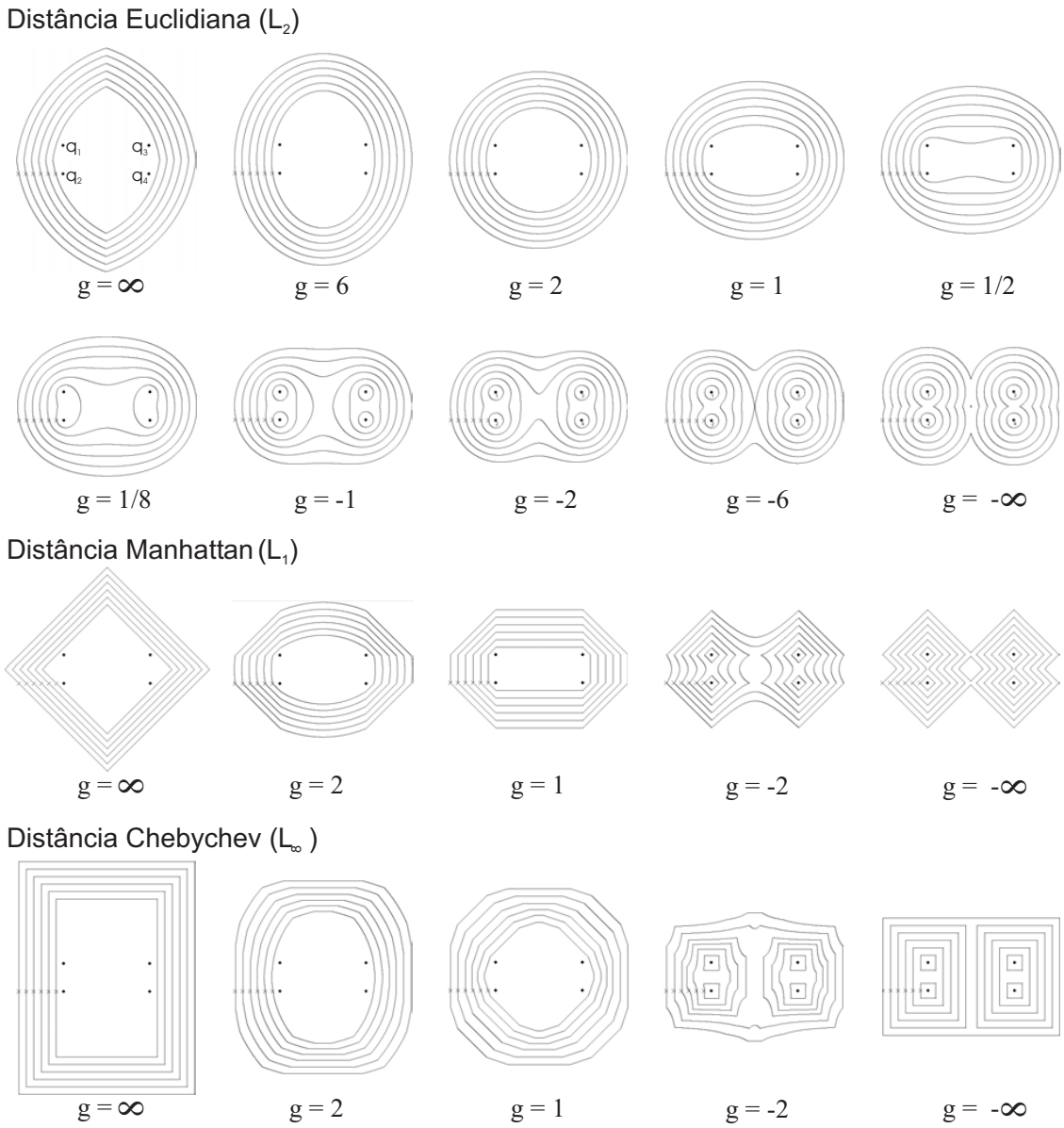

Figura 4.2: O efeito do fator de agregação g nos espaços Euclidiano, Manhattan e Chebychev de duas dimensões, considerando $Q = \{q_1, q_2, q_3, q_4\}$.

4.2 e 4.3 a seguir.

$$\begin{aligned} d_{g=\infty}(Q, s_i) = & \lim_{g \to \infty} \sqrt[g]{\sum_{s_q \in Q} \delta(s_q, s_i)^g} \\ = & \ max(\delta(s_q, s_i)), \quad \forall \ s_q \in Q \end{aligned} \tag{4.2}$$

$$d_{g=-\infty}(Q, s_i) = \lim_{g \to -\infty} \sqrt[g]{\sum_{s_q \in Q} \delta(s_q, s_i)^g}$$
$$= min(\delta(s_q, s_i)), \quad \forall \ s_q \in Q \tag{4.3}$$

A semântica do fator de agregação g está diretamente ligada com a descontinuidade (ou continuidade) semântica do par definido pela função de distância e pelo tipo de característica extraída de cada imagem. Por exemplo, ao se empregar $Q = \{q_1, q_2, q_3\}$ como na Figura 4.3 em um espaço euclidiano de duas dimensões e $g = 2$, que corresponde a minimização da distância média quadrática, a abrangência para um dado raio agregado tende ao centro gravitacional (ou baricentro para o caso da cardinalidade $|Q| = 3$) com relação ao conjunto de centros de consulta.

Figura 4.3: Abrangência da consulta com centros $Q = \{q_1, q_2, q_3\}$ e fator de agregação $g = 2$.

Entretanto, ao se considerar a descontinuidade semântica das características, pode-se adotar $g \to -\infty$ para se obter os elementos mais próximos de cada centro, com as distâncias representadas na Figura 4.4.

Figura 4.4: Abrangência da consulta com centros $Q = \{q_1, q_2, q_3\}$ e fator de agregação $g = -\infty$.

A Figura 4.5 apresenta a idéia básica da semântica associada à função de dissimilaridade agregada $d_g()$ baseada em um conjunto de características ideal de um conjunto de imagens e sua respectiva função de distância. Em (a) obtém-se o resultado a partir dos centros de consulta $Q = \{q_1, q_2\}$ e fator de agregação $g = -\infty$, que retorna imagens mais próximas de cada centro de consulta individualmente, representados pelas imagens dos Presidentes da República, Luiz Inácio Lula da Silva e Fernando Henrique Cardoso, sendo que o resultado é ordenado pela distância calculada em relação ao centro mais próximo de cada imagem. Em (b) obtém-se o resultado a partir dos centros de consulta

$Q = \{q_1, q_2\}$ e fator de agregação $g = 2$, que retorna imagens mais próximas considerando ambos os centros de consulta, e nesse exemplo, resulta na imagem que foi criada a partir de uma técnica de *morphing*[1], ou seja, uma imagem similar às duas imagens de consulta. Nesse caso, considera-se a continuidade semântica das características extraídas de um par de imagens, sendo que o conjunto de características extraídas da imagem resultado do *morphing* encontra-se próximo a ambas as imagens de consulta no espaço das características. Assim, a continuidade semântica dá-se quando, dadas duas imagens similares $\{A, B\}$ e uma pequena dissimilaridade dos seus respectivos mapeamentos para o espaço das características, ao se mapear para esse espaço uma terceira imagem C similar às imagens $\{A, B\}$, resulta em uma pequena dissimilaridade em relação às características de ambas as imagens $\{A, B\}$ nesse espaço.

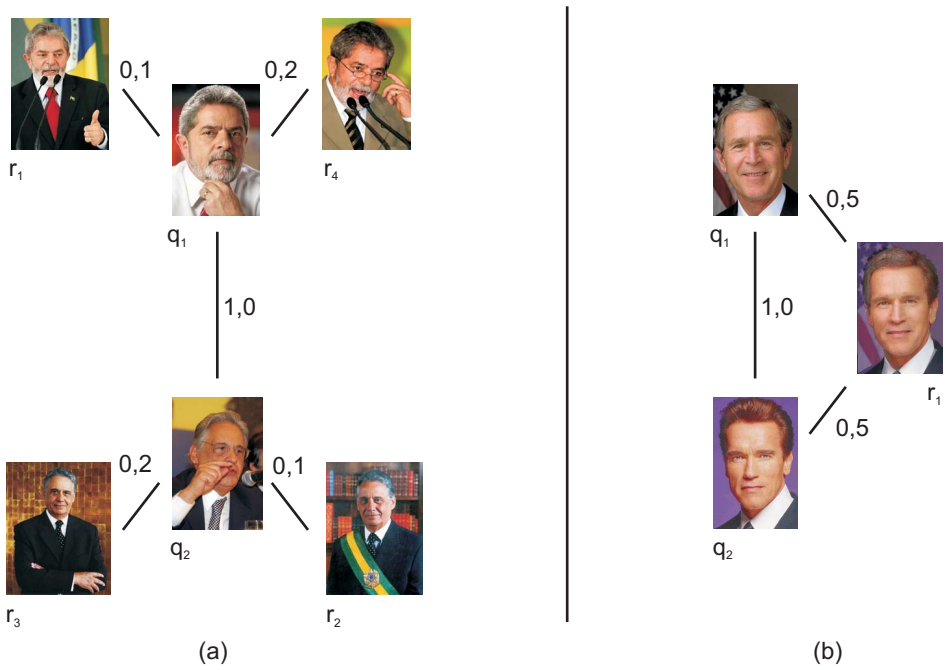

Figura 4.5: Idéia básica da semântica associada à função de dissimilaridade agregada $d_g()$. (a) $Q = \{q_1, q_2\}$ e fator de agregação $g = -\infty$. (b) $Q = \{q_1, q_2\}$ e fator de agregação $g = 2$.

4.1.1 Atribuição de Pesos

Conforme mencionado na Seção 3.3, a percepção dos usuários em métodos de realimentação de relevância, pode ser modelada por meio da atribuição de pesos maiores para as imagens que o usuário julgar mais importantes e pesos menores para as imagens menos importantes, além de pesos negativos para as imagens indesejadas. A Equação 4.1 permite a atribuição de pesos, que alteram a região de abrangência em uma consulta por

[1] *Morphing* ou *image warping* são técnicas utilizadas na criação de animações gráficas que permitem transformar uma imagem em outra, gerando um número de transições desejado.

abrangência agregada ou a relação das distâncias em uma consulta aos k-vizinhos mais próximos agregados. A Figura 4.6 ilustra o uso de pesos positivos e negativos, na qual é apresentada a região de abrangência em um espaço euclidiano de duas dimensões para um conjunto de três centros e fator de agregação $g = 1$. Nessa figura, a isolinha (a) representa a região de abrangência considerando os pesos $w_1 = w_2 = w_3 = 1$ (realimentação positiva) e a isolinha (b) representa a abrangência ao atribuir o peso $w_2 = -0.5$ (realimentação negativa) para o centro de consulta q_2 mantendo $w_1 = w_3 = 1$ para os centros de consulta q_1 e q_3. Nesse exemplo, é possível perceber que a realimentação negativa afastou a região de abrangência do elemento realimentado negativamente.

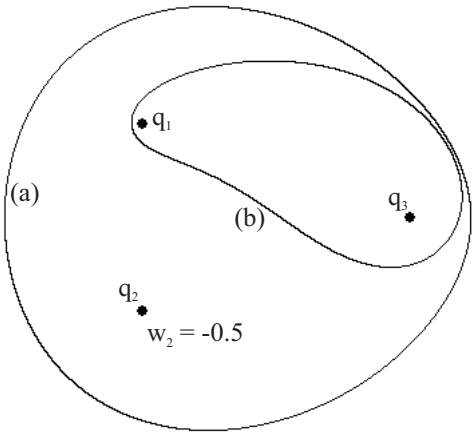

Figura 4.6: Região de abrangência em um espaço euclidiano de duas dimensões e fator de agregação $g = 1$ para o conjunto de centros $Q = \{q_1, q_2, q_3\}$. A isolinha (a) representa a abrangência considerando os pesos $w_1 = w_2 = w_3 = 1$ enquanto (b) representa a abrangência ao atribuir o peso $w_2 = -0.5$ para o centro de consulta q_2.

Definição 4. Uso de pesos negativos. Um peso negativo para expressar realimentação negativa na Equação 4.1 só deve ser empregada quando $g > 0$.

A atribuição de pesos negativos para expressar realimentação negativa na Equação 4.1 só deve ser empregada quando $g > 0$, uma vez que valores negativos de g tendem à função mínimo (Equação 4.3) e um peso negativo multiplicado por uma distância computada pela função δ sendo avaliado pela função mínimo levaria a um efeito contrário ao desejado, ou seja, a função de minimização escolheria os maiores valores, uma vez que esses estariam multiplicados por um valor negativo, ao invés de escolher os menores valores.

4.1.2 Propriedade do Raio Agregado Mínimo

Uma propriedade das consultas por abrangência agregada é que existe um valor para o raio agregado mínimo de modo a se obter uma região de abrangência não nula. Em uma consulta por abrangência de único centro, qualquer raio $\xi \geq 0$ a partir de um elemento de consulta determina uma região de abrangência não nula que pode ou não abranger

elementos de um conjunto de dados, e no caso do raio $\xi = 0$ permite a realização da consulta pontual. Entretanto, para as consultas de múltiplos centros, o raio agregado mínimo ξ para definir uma região de abrangência não nula pode ser um valor maior que zero. Esse valor depende da cardinalidade do conjunto de centros de consulta, das distâncias entre eles e do fator de agregação g.

Definição 5. Raio agregado mínimo. Dado um conjunto de elementos de consulta Q e um fator de agregação $g \neq 0$, existe um raio agregado mínimo $\xi_m \geq 0$ que define uma região não nula na qual uma consulta pode encontrar elementos s_i de um conjunto de dados.

Ao realizar uma consulta por abrangência agregada com raio agregado ξ menor que o raio agregado mínimo ξ_m, o resultado será um conjunto vazio independentemente da distribuição dos elementos no conjunto de dados. A título de ilustração e sem perda de generalidade, considere uma consulta por abrangência agregada baseada no conjunto de centros de consulta $Q = \{q_1, q_2, q_3, q_4\}$ como apresentado na Figura 4.7, de modo que h_1 é o elemento que minimiza a função de dissimilaridade agregada com relação à Q.

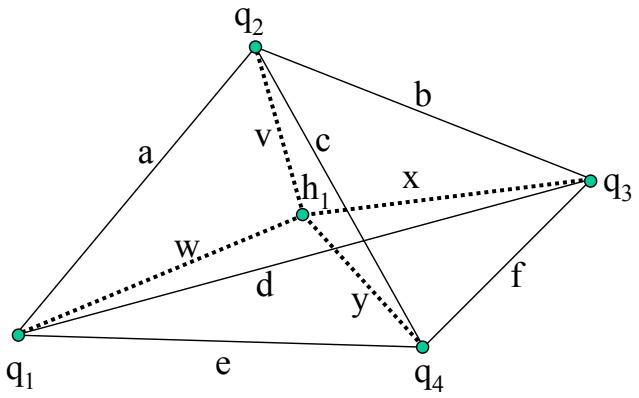

Figura 4.7: Ilustração do raio agregado mínimo.

Na figura, $\{a, b, c, d, e, f\}$ representam as distâncias conhecidas entre todos os centros de consulta. Entretanto, como o elemento h_1 não é conhecido (e possivelmente nem existe), as distâncias $\{v, w, x, y\}$ não podem ser computadas. Assim, se a dissimilaridade agregada $d_g(Q, h_1)$ representa o raio agregado mínimo ξ_m, computar ξ_m pode ser definido pela minimização de $d_g(Q, h_1) = \sqrt[g]{v^g + w^g + x^g + y^g}$. O problema de minimização para o exemplo apresentado na Figura 4.7 pode ser definido do seguinte modo. No exemplo, as distâncias $\{v, w, x, y\}$ consideram os seus pesos associados $\{w_v, w_w, w_x, w_y\}$: $v = w_v \cdot \delta(q_2, h_1)$, $w = w_w \cdot \delta(q_1, h_1)$, $x = w_x \cdot \delta(q_3, h_1)$ e $y = w_y \cdot \delta(q_4, h_1)$.

Lema 1. A minimização da função $d_g(Q, h_1) = \sqrt[g]{v^g + w^g + x^g + y^g}$ sujeito às inequações baseadas na propriedade de desigualdade triangular apresentadas na Equação 4.4 determina o raio agregado mínimo.

$$\begin{array}{ccccccccccc}
\mathrm{v} & + & \mathrm{w} & & & & & - & \mathrm{a} & \geq & 0 \\
\mathrm{v} & & & + & \mathrm{x} & & & - & \mathrm{b} & \geq & 0 \\
\mathrm{v} & & & & & + & \mathrm{y} & - & \mathrm{c} & \geq & 0 \\
 & & \mathrm{w} & + & \mathrm{x} & & & - & \mathrm{d} & \geq & 0 \\
 & & \mathrm{w} & & & + & \mathrm{y} & - & \mathrm{e} & \geq & 0 \\
 & & & & \mathrm{x} & + & \mathrm{y} & - & \mathrm{f} & \geq & 0
\end{array} \tag{4.4}$$

Os fatores de agregação $g = 1$ (minimizar $d_g(Q, h_1) = v + w + x + y$), $g = \infty$ (minimizar $d_g(Q, h_1) = \mathit{máximo}(v, w, x, y)$) e $g = -\infty$ (minimizar $d_g(Q, h_1) = \mathit{mínimo}(v, w, x, y)$) sujeito às inequações apresentadas na Equação 4.4 resultam no problema de encontrar os mínimos de funções multidimensionais arbitrárias, conhecido como métodos de programação linear. O problema de minimização multidimensional requer encontrar um ponto h_1 no qual a função escalar $f(q_1, q_2, ..., q_n)$ resulta no menor valor que qualquer vizinho de h_1, sendo que n corresponde à cardinalidade de $|Q|$. Um exemplo de algoritmo de minimização multidimensional é o algoritmo Simplex [Nelder e Mead, 1965], que mantém $n + 1$ vetores de parâmetros parciais como os vértices de um simplex n-dimensional. A cada iteração o algoritmo tenta melhorar o pior vértice por meio de uma transformação geométrica simples até o tamanho do simplex ser menor que uma dada tolerância [Bazaraa et al., 2004, Galassi et al., 2006]. Para outros valores do fator de agregação g, técnicas de programação não-linear – como o método de Newton [Luenberger, 1984] – podem ser usadas para computar o raio agregado mínimo ξ_m.

O número de inequações para as quais a função de minimização está sujeita é a combinação sem repetição dos pares de elementos composto pelos centros de consulta Q, que é dado pelo binômio definido na Equação 4.5:

$$\left(\begin{array}{c} n \\ m \end{array} \right) = \frac{n!}{m! \cdot (n - m)!} \tag{4.5}$$

no qual n é o número de centros de consulta e m é o número de valores desconhecidos que são combinados usando a propriedade de desigualdade triangular ($m = 2$). Por exemplo, um conjunto de 4 centros de consulta resulta em 6 inequações (como mostrado no exemplo da Figura 4.7), um conjunto de 5 centros resulta em 10 inequações e assim por diante.

Em geral, sistemas de equações lineares e não lineares podem resultar em sistemas inconsistentes, ou seja, quando não existe ao menos uma solução admissível para o problema. Um sistema inconsistente é resultado de contradições entre as restrições. É importante notar que o sistema apresentado no Lema 1 resulta em um sistema consistente.

Teorema 1. A função de minimização que determina a propriedade do raio agregado mínimo ξ_m sujeita ao conjunto de inequações derivadas da propriedade de desigualdade

triangular sempre resulta em um sistema consistente.

Prova. Baseado no exemplo da Figura 4.7 e sem perda de generalidade, as constantes $\{a, b, c, d, e, f\}$ e variáveis $\{v, w, x, y\}$ envolvidas são maiores ou iguais a zero, uma vez que são resultados de uma função de distância e logo respeitam a propriedade de não-negatividade. Assim, dado que:

$$0 \leq v \leq max\{a, b, c\} \text{ e}$$
$$0 \leq w \leq max\{a, d, e\} \text{ e}$$
$$0 \leq x \leq max\{b, d, f\} \text{ e}$$
$$0 \leq y \leq max\{c, e, f\},$$

não existe um conjunto de valores para as variáveis $\{v, w, x, y\}$ que resulta em um sistema inconsistente. □

4.2 Otimização

A execução de consultas usando múltiplos centros em um MAM não é trivial, uma vez que os elementos são organizados em relação a um elemento de referência por página de disco, e não a múltiplos elementos. Para que seja possível incorporar técnicas de realimentação de relevância para recuperação de imagens por conteúdo em sistemas de gerenciamento de banco de dados, é preciso que a computação de consultas usando múltiplos centros seja realizada com algoritmos e estruturas de dados adequados e eficientes.

Para realizar consultas por similaridade, para cada tipo de dados deve existir uma função de distância $\delta()$ que atenda as propriedades dos espaços métricos de identidade, simetria, não-negatividade e desigualdade triangular. Conforme descrito na Seção 2.4.2, os métodos de acesso métrico são estruturas de indexação hierárquicas apropriadas para organizar dados que são consultados por similaridade, sendo que em geral foram desenvolvidos para melhorar o desempenho de consultas por similaridade de único centro, como as consultas por abrangência e aos k-vizinhos mais próximos. Esses métodos usam a propriedade de desigualdade triangular para evitar leituras e comparações com ramos das hierarquias (vide Seção 2.4.3).

Em se tratando de consultas aos k-vizinhos mais próximos, vários métodos para otimização de seu desempenho foram propostos, entre eles o método *branch-and-bound* [Roussopoulos et al., 1995], o método incremental [Hjaltason e Samet, 1995] e os algoritmos *multi-step* [Korn et al., 1996, Seidl e Kriegel, 1998]. Todos esses trabalhos referem-se a algoritmos que lidam com apenas um centro de consulta.

O uso de agregação de distâncias em consultas por similaridade foi primeiramente

usado como mecanismo de realimentação de relevância para recuperação de imagens por conteúdo no sistema denominado Falcon descrito na Seção 2.4.3. O algoritmo de busca proposto consiste na união das consultas de único centro realizadas para em cada centro de consulta com um raio ϵ, seguida de uma etapa de filtragem que avalia se cada elemento resultante da união possui similaridade agregada menor ou igual a raio agregado da consulta ξ. Entretanto, essa solução depende de um parâmetro ϵ empírico, na qual a semântica depende da noção de similaridade do conjunto pelo usuário.

Com relação às consultas limitadas por k, o *Minimum Bounding Method* (MBM), descrito na Seção 2.4.3, é baseado em uma função de agregação para ordenar dados espaciais com baixo número de dimensões para resolver consultas aos k-vizinhos mais próximos agregados. Por ter sido desenvolvido para dados espaciais, esse método utiliza propriedades geométricas para realizar podas em estruturas do tipo R-tree. Assim, o MBM não pode ser generalizado para dados em espaços métricos, uma vez que uma parte da solução está intimamente ligada à função de distância empregada.

4.2.1 Metric Aggregate Similarity Search (MASS)

As consultas apresentadas na Seção 4.1 permitem a ordenação dos elementos de um conjunto de dados pela ordem crescente das distâncias agregadas (Equação 4.1) em relação ao conjunto de centros de consulta Q. Nesta seção é apresentada a técnica *Metric Aggregate Similarity Search (MASS)* para resolver as consultas por similaridade agregada que utilizam a função de agregação, considerando seis dos valores mais significativos do fator de agregação $g \in \{-\infty, -2, 1/2, 1, 2, \infty\}$. A Figura 4.8 apresenta um exemplo de consulta por abrangência agregada em um espaço bidimensional euclidiano e $g = 1$. Considere uma consulta com centros $\{q_1, q_2\}$ e uma sub-árvore com centro em s_t e raio de cobertura r_t. Considere o elemento hipotético h_1 como o elemento com a maior distância possível do representante que poderia ser armazenado na sub-árvore e que minimizaria $d_g()$ com relação a q_1 e q_2. Embora a Figura 4.8 tenha sido desenhada utilizando a distância euclidiana por ser mais intuitiva, na discussão abaixo, todas as distâncias $(a, c, r_1, t, u, v\ , w)$, são independentes da função de distância real $\delta()$ e do fator de agregação g empregado. O desafio é computar o limite inferior da similaridade agregada dos centros q_1 e q_2 até h_1 para decidir se o raio agregado (região da consulta) intersecta a região coberta pela sub-árvore.

Conforme a Figura 4.8, r_t é conhecido e $\{a, c\}$ podem ser calculados, mas $\{v, w\}$ não podem. Para garantir que uma sub-árvore centrada em s_t com raio de cobertura r_t possa ser podada, é preciso determinar se $d_g(Q, h_1) = \sqrt[g]{v^g + w^g}$ é menor ou igual ao raio agregado limitante $\xi = \sqrt[g]{t^g + u^g}$ que gera a região de consulta, ou seja, se as duas regiões intersectam-se. Se não houver intersecção, a sub-árvore centrada em s_t pode

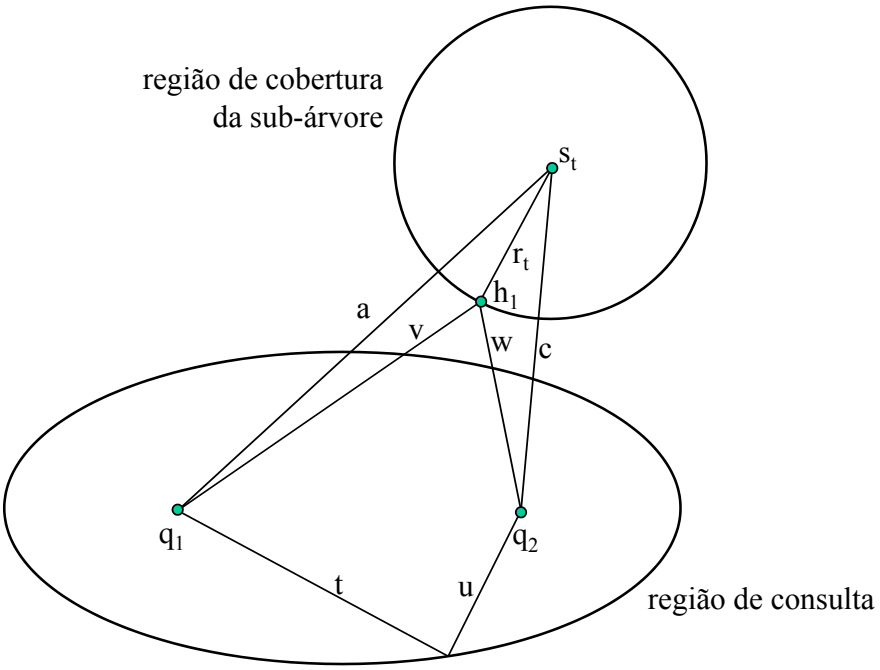

Figura 4.8: Consulta por abrangência agregada em um espaço bidimensional euclidiano, $g = 1$, $\{q_1, q_2\}$ formam o conjunto de centros Q, s_t é um elemento representante de uma sub-árvore e r_t é o raio de cobertura dessa sub-árvore.

ser podada. Pela definição de funções de distância em um espaço métrico, as seguintes equações baseadas na desigualdade triangular são verdadeiras:

$$\begin{aligned} a &\leq r_t + v \\ c &\leq r_t + w \end{aligned} \tag{4.6}$$

Isolando v e w tem-se:

$$\begin{aligned} v &\geq |a - r_t| \\ w &\geq |c - r_t| \end{aligned} \tag{4.7}$$

Logo, para $g = 1$, $d_g = v + w$:

$$\begin{aligned} v &\geq |a - r_t| \\ w &\geq |c - r_t| \\ v + w &\geq |a - r_t| + |c - r_t| \end{aligned} \tag{4.8}$$

E para $g = 2$, $d_g = \sqrt[2]{v^2 + w^2}$, como no caso anterior:

$$\begin{aligned} v^2 &\geq |a - r_t|^2 \\ w^2 &\geq |c - r_t|^2 \\ v^2 + w^2 &\geq |a - r_t|^2 + |c - r_t|^2 \end{aligned} \tag{4.9}$$

Generalizando, pode ser definido que, para $g \neq 0$:

$$\begin{aligned} v^g &\geq |a - r_t|^g \\ w^g &\geq |c - r_t|^g \\ v^g + w^g &\geq |a - r_t|^g + |c - r_t|^g \end{aligned} \tag{4.10}$$

Os exemplos a seguir apresentam casos especiais para outros valores significativos de g. Para $g = 1/2$, $d_g = (\sqrt[2]{v} + \sqrt[2]{w})^2$:

$$\begin{aligned} \sqrt[2]{v} &\geq \sqrt[2]{|a - r_t|} \\ \sqrt[2]{w} &\geq \sqrt[2]{|c - r_t|} \\ \sqrt[2]{v} + \sqrt[2]{w} &\geq \sqrt[2]{|a - r_t|} + \sqrt[2]{|c - r_t|} \end{aligned} \tag{4.11}$$

E para valores negativos de g, a relação também é verdadeira. Para $g = -2$, $d_g = \sqrt[-2]{v^{-2} + w^{-2}} = \sqrt[-2]{\frac{v^2+w^2}{v^2 \cdot w^2}}$:

$$\begin{aligned} v^{-2} &\geq |a - r_t|^{-2} \\ w^{-2} &\geq |c - r_t|^{-2} \\ \frac{v^2 + w^2}{v^2 \cdot w^2} &\geq \frac{|a - r_t|^2 + |c - r_t|^2}{|a - r_t|^2 \cdot |c - r_t|^2} \end{aligned} \tag{4.12}$$

Outros exemplos incluem $g = \infty$, $d_g = \lim_{g \to \infty}(\sqrt[g]{v^g + w^g}) = max(v, w)$:

$$\begin{aligned} v^\infty &\geq |a - r_t|^\infty \\ w^\infty &\geq |c - r_t|^\infty \\ max(v, w) &\geq max(|a - r_t|, |c - r_t|) \end{aligned} \tag{4.13}$$

e $g = -\infty$, $d_g = \lim_{g \to -\infty}(\sqrt[g]{v^g + w^g}) = min(v, w)$:

$$
\begin{aligned}
v^{-\infty} &\geq |a - r_t|^{-\infty} \\
w^{-\infty} &\geq |c - r_t|^{-\infty} \\
min(v, w) &\geq min(|a - r_t|, |c - r_t|)
\end{aligned}
\tag{4.14}
$$

O raio agregado ξ é um parâmetro da consulta. A combinação das restrições acima permitem definir um algoritmo para computar a sobreposição entre uma região de consulta coberta por um raio agregado com relação a Q e uma sub-árvore centrada em s_t com raio de cobertura r_t. A computação da sobreposição é apresentada a seguir no Algoritmo 2.

Algoritmo 2 MASS – Verifica se há intersecção entre uma sub-árvore e a região de consulta.

Entrada: conjunto de centros de consulta Q, raio agregado da consulta ξ, representante da sub-árvore s_t, raio de cobertura r_t, fator de agregação g

1: **se** $\xi \geq \sqrt[g]{\sum_{s_q \in Q} \mid \delta(s_q, s_t) - r_t \mid^g}$ **então**
2: retornar **verdadeiro**
3: **senão**
4: retornar **falso**
5: **fim se**

4.2.1.1 Consultas por Abrangência Agregada

O algoritmo proposto para consultas por abrangência agregada emprega a estratégia de travessia em profundidade e em largura, como nos algoritmos tradicionais para consultas por abrangência em métodos de acesso métrico como a M-tree e a Slim-tree. A partir do nodo raiz, navega-se recursivamente pela hierarquia, verificando se cada sub-árvore pode ser podada. O Algoritmo 3 apresenta a consulta por abrangência agregada, que recebe como parâmetros um conjunto de centros de consulta Q, um valor para o fator de agregação g e um raio agregado da consulta ξ, e retorna a lista *resultado* ordenada pela similaridade, contendo os elementos $s_i \in S$ armazenados no índice que atendem aos critérios da consulta. A lista *resultado* é ordenada pela similaridade agregada entre cada elemento do conjunto e o conjunto de centros de consulta. No passo 3, o Algoritmo 3 executa o Algoritmo 4, percorrendo a estrutura recursivamente.

O passo 2 do Algoritmo 4 executa a travessia em profundidade na hierarquia. O Algoritmo 2 é chamado no passo 3 para decidir pela poda da sub-árvore em questão. O passo 4 realiza a chamada recursiva no caso de ser necessário analisar a sub-árvore apontada por $nodo.entrada_i$. Quando um nodo folha é alcançado (passo 8), os elementos que atendem ao critério da consulta são adicionados no conjunto resultado. No passo 9, a

Algoritmo 3 ARq – Consulta por abrangência agregada em um índice métrico.

Entrada: conjunto de centros de consulta Q, fator de agregação g, raio agregado da consulta ξ

1: $resultado = novaListaDeElementos()$
2: $nodo = PaginaRaizDoIndice()$
3: $RecursãoARq(nodo.entrada_i, resultado, Q, g, \xi)$
4: retornar $resultado$

Algoritmo 4 RecursãoARq – Recursão da consulta por abrangência agregada em um índice métrico.

Entrada: raíz da sub-árvore *nodo*, lista *resultado*, conjunto de centros de consulta Q, fator de agregação g, raio agregado da consulta ξ

1: **se** *nodo*.Tipo = *indice* **então**
2: **para todo** elemento representante $s_i \in$ nodo **faça**
3: **se** MASS$(Q, \xi, nodo.s_i, nodo.raio_i, g)$ **então**
4: $RecursãoARq(nodo.entrada_i, resultado, Q, g, \xi)$ // Recursão
5: **fim se**
6: **fim para**
7: **senão se** *nodo*.Tipo = $folha$ **então**
8: **para todo** elemento $s_i \in$ nodo **faça**
9: $d = d_g(g, Q, nodo.s_i)$
10: **se** $d \leq \xi$ **então**
11: $resultado$.Adicionar(d, $nodo.s_i$)
12: **fim se**
13: **fim para**
14: **fim se**

Equação 4.1 é usada para computar a distância agregada de um elemento para o conjunto de centros de consulta Q.

4.2.1.2 Consultas aos k-Vizinhos Mais Próximos Agregado

O Algoritmo 5 responde a consulta aos k-vizinhos mais próximos agregado proposta, que baseia-se na estratégia "o melhor primeiro" (*best-first*) empregada para algoritmos de busca aos vizinhos mais próximos [Roussopoulos et al., 1995], estendido para consultas de múltiplos centros. A idéia é usar os elementos retornados durante a travessia pela estrutura para reduzir dinamicamente o raio de consulta, ao qual inicialmente foi atribuído o valor ∞. O raio é então diminuído ao encontrar um novo elemento que tenha distância agregada menor que o maior valor na lista de resultados já encontrados. Esse raio é usado durante a travessia para podar sub-árvores que garantidamente não contêm elementos com distância agregada menor que o elemento com a maior distância agregada já inserido no conjunto de resultados. A lista de prioridade usada na implementação piloto do algoritmo é uma árvore binária de busca (AVL) que armazena os elementos de dados ordenados pela sua distância agregada computada a partir dos centros de consulta Q. Note que, no passo 8,

o algoritmo MASS é executado para decidir se uma sub-árvore é adicionada na lista de prioridades.

Algoritmo 5 k-ANNq – Consulta aos k-vizinhos mais próximos agregado em um índice métrico.

Entrada: conjunto de centros de consulta Q, fator de agregação g, número de elementos k

```
1: resultado = novaListaDeElementos()
2: fila = novaListaDePrioridades()
3: nodo = PaginaRaizDoIndice()
4: raio = ∞
5: enquanto fila.Vazia() = falso faça
6:    se nodo.Tipo = indice então
7:       para todo elemento representante s_i ∈ nodo faça
8:          se MASS(Q, raio, nodo.s_i, nodo.raio_i, g) então
9:             distância = d_g(g, Q, nodo.s_i)
10:            fila.Adicionar(distância, nodo.s_i)
11:         fim se
12:      fim para
13:   senão se nodo.Tipo = folha então
14:      para todo elemento s_i ∈ nodo faça
15:         distância = d_g(g, Q, nodo.s_i)
16:         se d ≤ raio então
17:            resultado.Adicionar(distância, nodo.s_i)
18:            se resultado.NúmeroDeElementos ≥ k então
19:               resultado.Remover(k)
20:               raio = resultado.DistânciaMáxima()
21:            fim se
22:         fim se
23:      fim para
24:   fim se
25:   nodo = fila.PróximoNodo()
26: fim enquanto
27: retornar resultado
```

4.3 Experimentos

Foram realizados experimentos para validação das consultas propostas em um método de realimentação de relevância e para validação da técnica de otimização. A seguir os experimentos são apresentados.

4.3.1 Consultas por Similaridade Agregada em um Método de Realimentação de Relevância

Para avaliar a adequação das consultas por similaridade agregada como uma técnica eficaz para melhorar o resultado de consultas por similaridade, efetuamos uma bateria de testes controlados. Para a realização dos experimentos, utilizou-se a base de imagens *Amsterdam Library of Object Images* (ALOI) [Geusebroek et al., 2005], uma coleção de imagens de 1.000 pequenos objetos, criada para fins científicos em várias configurações. A configuração utilizada é a *ALOI Illumination Color*, na qual foram aplicadas 12 diferentes cores de iluminação aos objetos, gerando 12.000 imagens. Foram extraídos histogramas de níveis de cinza quantizados em 256 níveis, como a média das intensidades das cores vermelho, verde e azul, considerando os pesos 0,2989, 0,5870 e 0,1141 respectivamente, relativos à sensibilidade/percepção humana com relação a essas três cores [Wright, 1929, Guild, 1931]. A função de distância empregada é a Manhattan (L_1). A Figura 4.9 apresenta uma amostra de 10 elementos desse conjunto de dados.

Figura 4.9: Amostra de imagens do conjunto *ALOI*.

O protótipo desenvolvido permite a execução de consultas aos k-vizinhos mais próximos e a realimentação da relevância das imagens resultantes. Após a realimentação do usuário, pode-se executar uma nova consulta considerando a relevância, e nesse caso podem ser empregadas as consultas por vizinhos mais próximos agregados ou a fórmula de Rocchio para a movimentação do centro de consulta. O efeito do fator de agregação g é avaliado com relação aos gráficos de precisão e revocação de consultas aos 40-vizinhos mais próximos e os respectivos 3 primeiros ciclos de realimentação de relevância. É importante lembrar que, em um gráfico de precisão e revocação, quanto mais próximo uma curva estiver do topo do gráfico, melhor é o resultado do método em avaliação (veja Seção 2.4.4). Nesses experimentos, são apresentados os resultados baseados apenas em consultas aos k-vizinhos mais próximos agregados dado que essas consultas são mais relevantes para os métodos de realimentação de relevância. Todos os ciclos de realimentação foram executados com consultas aos 40-vizinhos mais próximos, sendo que cada objeto do mundo real é considerado como uma classe distinta no cálculo da precisão e para cada consulta,

as imagens realimentadas como relevantes são as imagens retornadas da consulta anterior que contém a mesma classificação da imagem de consulta. Os gráficos representam a média dos resultados para 100 consultas escolhidas aleatoriamente.

4.3.1.1 Realimentação Positiva

Neste experimento apenas a realimentação positiva é avaliada, com variação do fator de agregação g com valores $g = \infty$, $g = 2$, $g = 1$, $g = -1$, $g = -2$ e $g = -\infty$. A Figura 4.10 apresenta os gráficos de precisão e revocação resultantes. A realimentação aumentou a precisão nos 3 primeiros ciclos em todos os gráficos, exceto para $g = \infty$. Esse comportamento pode ser explicado pela descontinuidade semântica das características extraídas das imagens nesse experimento em relação à percepção humana, uma vez que valores de $g = \infty$ resultam em regiões localizadas entre os elementos de consulta, considerando a continuidade semântica dos vetores resultantes. Após os 3 primeiros ciclos, comparando os resultados para 76% de revocação, foram obtidas as precisões de 44% para $g = \infty$, 65% para $g = 2$, 69% para $g = 1$, 71% para $g = -1$, 75% para $g = -2$ e 80% para $g = -\infty$. É importante notar que valores elevados de precisão em níveis de revocação altos são resultados desejáveis e que nesse experimento a precisão aumentou com a diminuição do fator de agregação g (de 44% para 80% com g variando de ∞ para $-\infty$ respectivamente), como era esperado para histogramas de níveis de cinza extraído das imagens.

O mesmo experimento foi executado com a técnica de movimentação do centro de consulta denominada fórmula de Rocchio, descrita na Seção 3.2.2. Para considerar apenas a realimentação positiva na fórmula de Rocchio, foram selecionadas as constantes $\alpha = 1$ (peso do elemento de consulta) e $\gamma = 0$ (peso da realimentação negativa). O experimento considerou a constante β (peso da realimentação positiva) com variação entre 0,2 e 1,2.

Para cada elemento de consulta, foi executada uma consulta aos 40-vizinhos mais próximos e em seguida foi realizada a movimentação do centro de consulta com base nos elementos realimentados como relevantes, e em seguida a consulta foi submetida novamente (ciclo de realimentação de relevância). Esse processo foi repetido por 3 ciclos. As configurações permitem a comparação direta dos resultados com os resultados das consultas apresentadas na Figura 4.10, uma vez que foram executadas nas mesmas condições. Note-se que a curva original dos 40-vizinhos mais próximos é a mesma em todos os gráficos das Figuras 4.10 e 4.11, servindo como referência para comparação. A Figura 4.11 apresenta os gráficos de precisão e revocação utilizando-se a fórmula de Rocchio. Nos gráficos, é possível verificar que a precisão aumentou com a aproximação do parâmetro β à 1, o que sugere que para esse experimento as imagens realimentadas como relevantes devem ter a mesma importância na movimentação do centro de consulta quanto à imagem de consulta utilizada. Utilizando-se a fórmula de Rocchio foram obtidos precisão de 61.6%

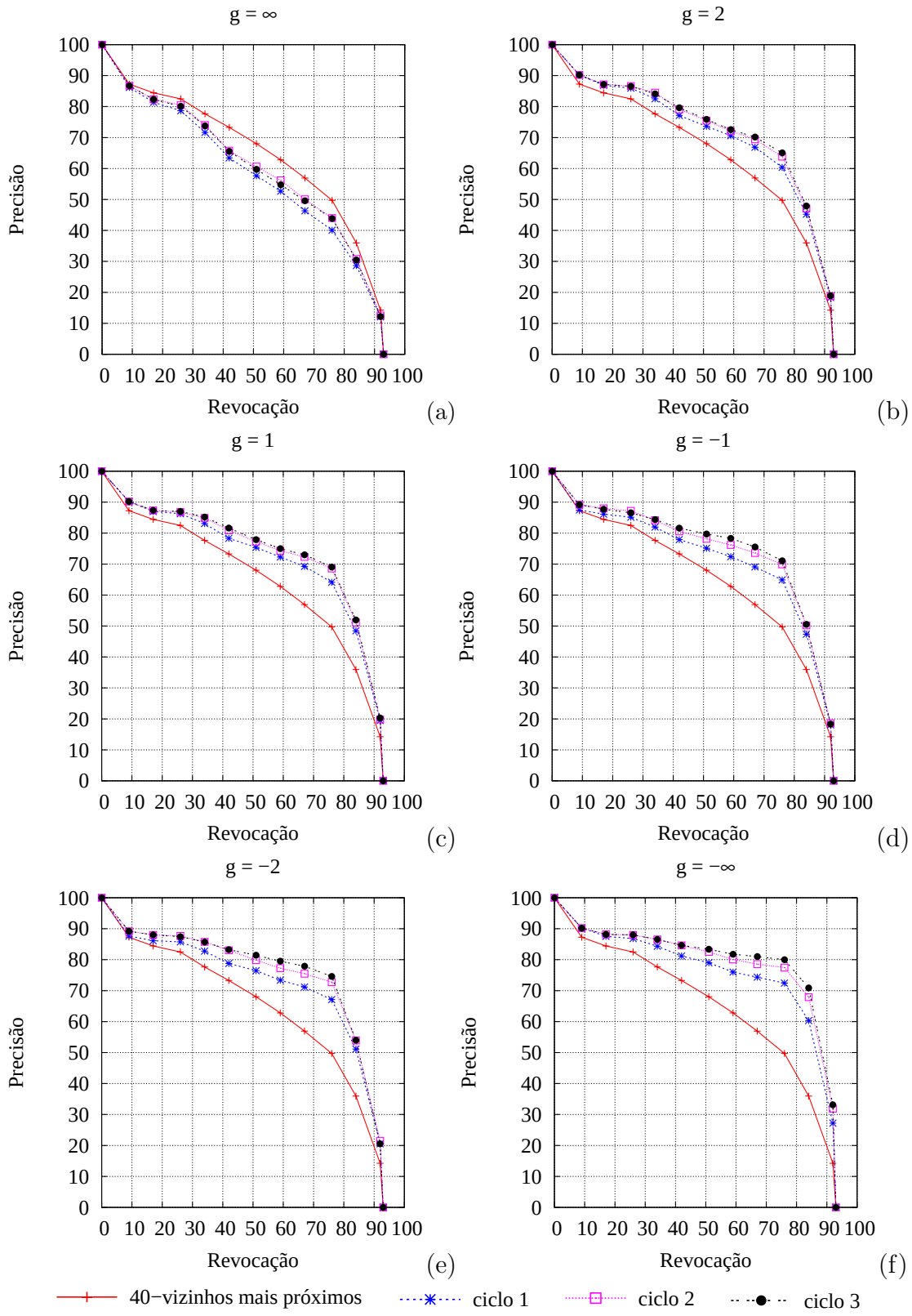

Figura 4.10: Realimentação positiva: gráficos de precisão e revocação para as consultas aos 40-vizinhos mais próximos e os 3 primeiros ciclos de realimentação utilizando a consulta aos k-vizinhos mais próximos agregados pelo método MASS. (a) $g = \infty$ (b) $g = 2$ (c) $g = 1$ (d) $g = -1$ (e) $g = -2$ (f) $g = -\infty$.

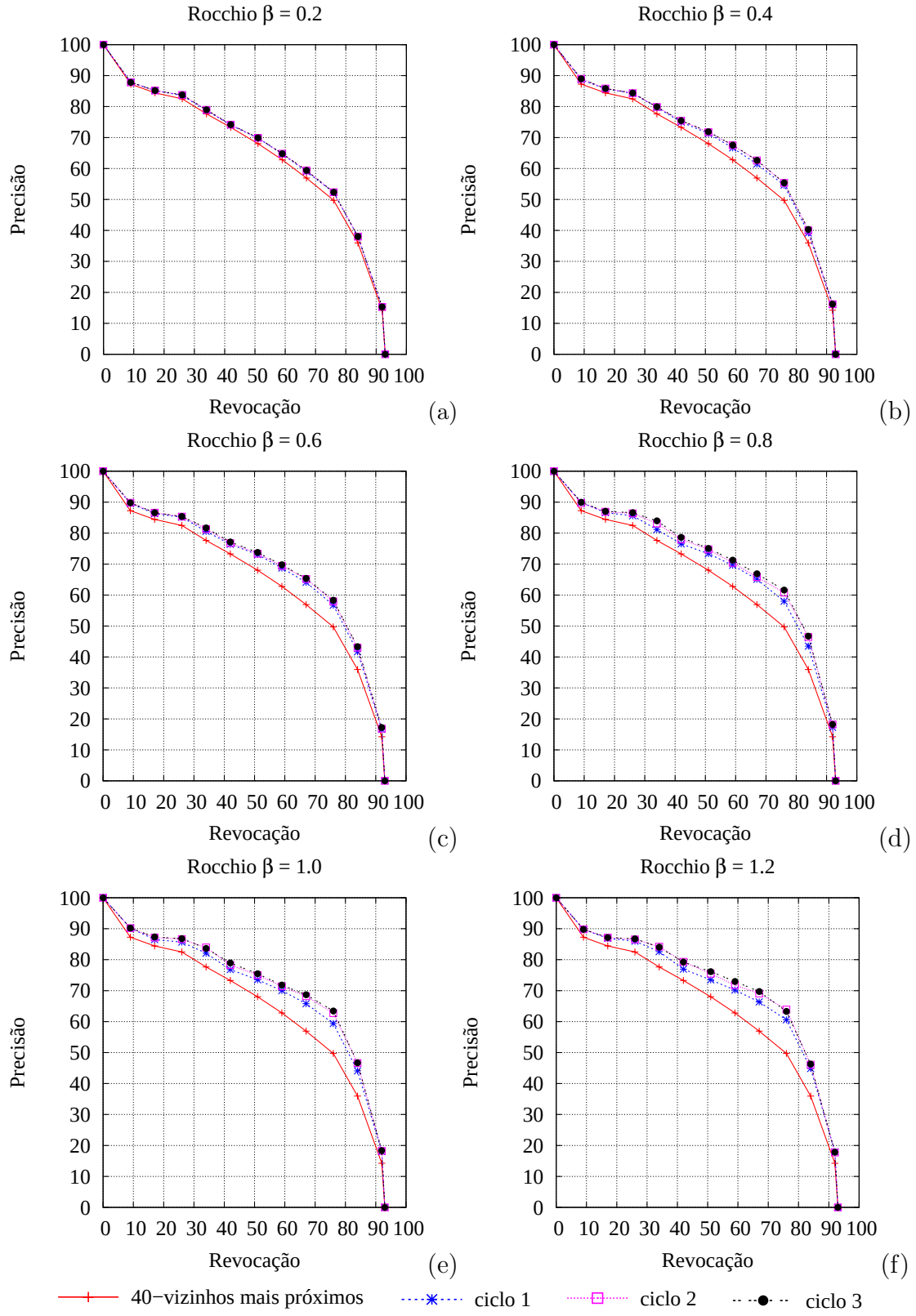

Figura 4.11: Realimentação positiva: gráficos de precisão e revocação para as consultas aos 40-vizinhos mais próximos e os 3 primeiros ciclos de realimentação utilizando a fórmula de Rocchio. (a) $\beta = 0,2$ (b) $\beta = 0,4$ (c) $\beta = 0,6$ (d) $\beta = 0,8$ (e) $\beta = 1,0$ (f) $\beta = 1,2$.

para $\beta = 0{,}8$, 63,5% para $\beta = 1$ e 63,3% para $\beta = 1{,}2$, considerando 76% de revocação. Assim, verifica-se que a fórmula de Rocchio atingiu uma precisão máxima na faixa de 63%, enquanto que o método MASS proposto chegou a até 80% de ganho de precisão para $g = -infty$ nesse experimento.

4.3.1.2 Realimentação Positiva e Negativa

Nesse experimento são consideradas a realimentação positiva e negativa, com variação do fator de agregação g com valores $g = \infty$, $g = 2$, $g = 1$, $g = -1$, $g = -2$ e $g = -\infty$. Para a realimentação negativa, foram escolhidas aleatoriamente, do resultado de cada consulta, um número de imagens de classe diferente da imagem de consulta na razão de 1/3 do número de imagens selecionadas como realimentação positiva, utilizando-se o peso $w = -1/2$ para as imagens realimentadas negativamente e peso $w = 1$ para as imagens realimentadas positivamente. A Figura 4.12 apresenta os gráficos de precisão e revocação resultantes. Considerando $g = 2$ e $g = 1$, houve aumento na precisão com a realimentação positiva e negativa, para 58% e 62% respectivamente para o nível de revocação de 76%, entretanto esse aumento foi inferior ao aumento na precisão considerando-se apenas a realimentação positiva. Como era previsto, a realimentação considerando $g = \infty$ não resultou em melhora de precisão. Para $g < 0$, a realimentação negativa gera o efeito de aproximar o resultado dos elementos classificados como negativos, ao invés de afastar desses elementos, uma vez que um peso negativo associado à função de distância sendo avaliado pela função mínimo leva a função de minimização à escolher as maiores distâncias, conforme descrito na Seção 4.1.1.

O mesmo experimento foi executado com a fórmula de Rocchio, descrita na Seção 3.2.2. Para tanto, foram selecionadas as constantes $\alpha = 1$ (peso do elemento de consulta) e $\beta = 1$ (peso da realimentação positiva). O experimento considerou a constante γ (peso da realimentação negativa) com variação entre 0,2 e 1,2. A Figura 4.13 apresenta os gráficos de precisão e revocação. Os resultados apresentam melhora na precisão quando os pesos da realimentação negativa $\gamma \leq 0{,}8$, indicando que nesses experimentos a realimentação negativa não contribui para a melhora da precisão. Para revocação de 75%, foram obtidas as precisões de 64% para $\gamma = 0{,}2$, 63% para $\gamma = 0{,}4$, 60% para $\gamma = 0{,}6$, 55% para $\gamma = 0{,}8$, 51% para $\gamma = 1{,}0$ e 43% para $\gamma = 1{,}2$.

4.3.2 Otimização

Os experimentos para avaliar a eficiência dos algoritmos foram realizados comparando a execução das consultas por similaridade agregada em um método de acesso seqüencial, os métodos disponíveis na literatura e os algoritmos propostos implementados no método

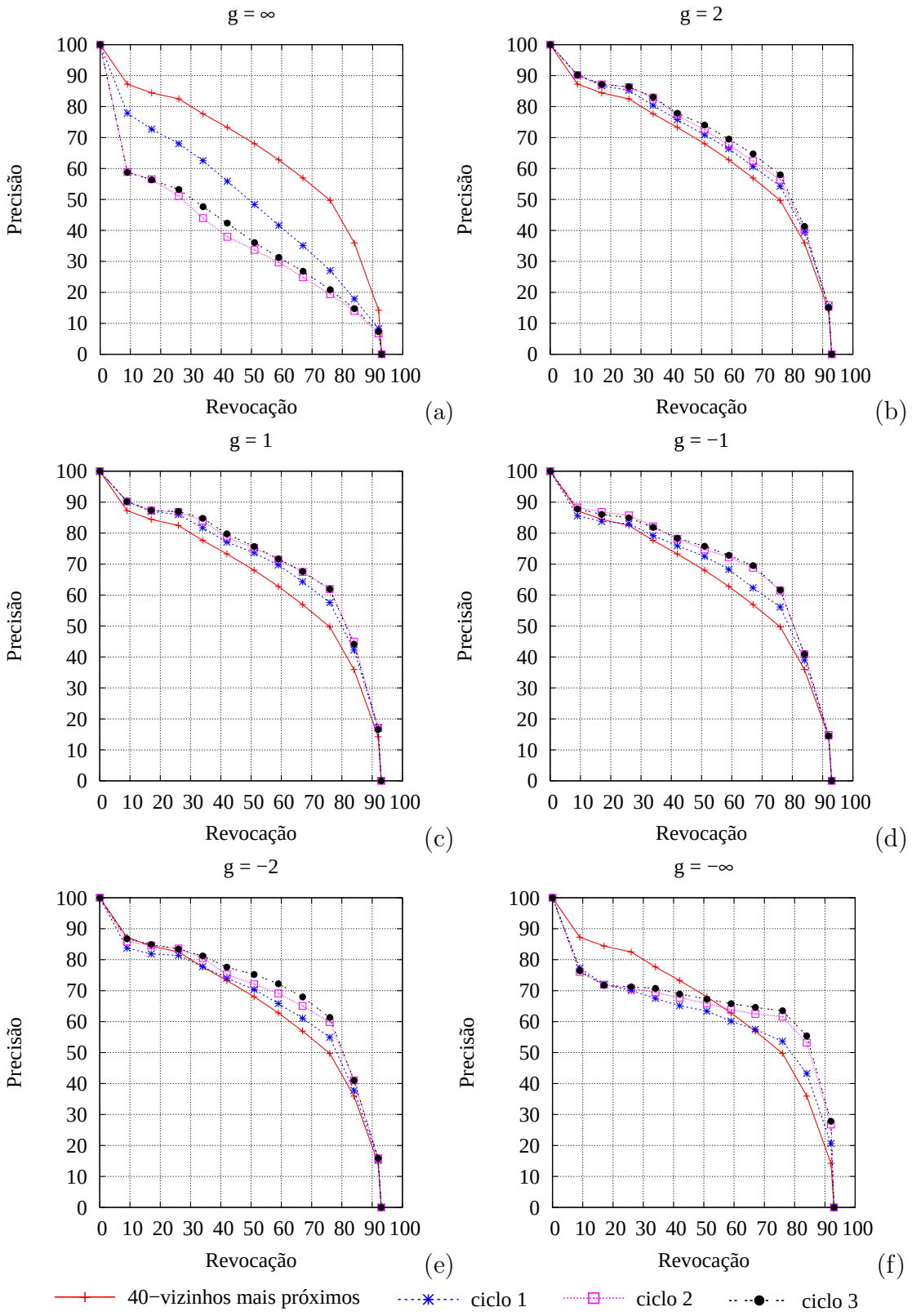

Figura 4.12: Realimentação positiva e negativa: gráficos de precisão e revocação para as consultas aos 40-vizinhos mais próximos e os 3 primeiros ciclos de realimentação utilizando a consulta aos k-vizinhos mais próximos agregados pelo método MASS. (a) $g = \infty$ (b) $g = 2$ (c) $g = 1$ (d) $g = -1$ (e) $g = -2$ (f) $g = -\infty$.

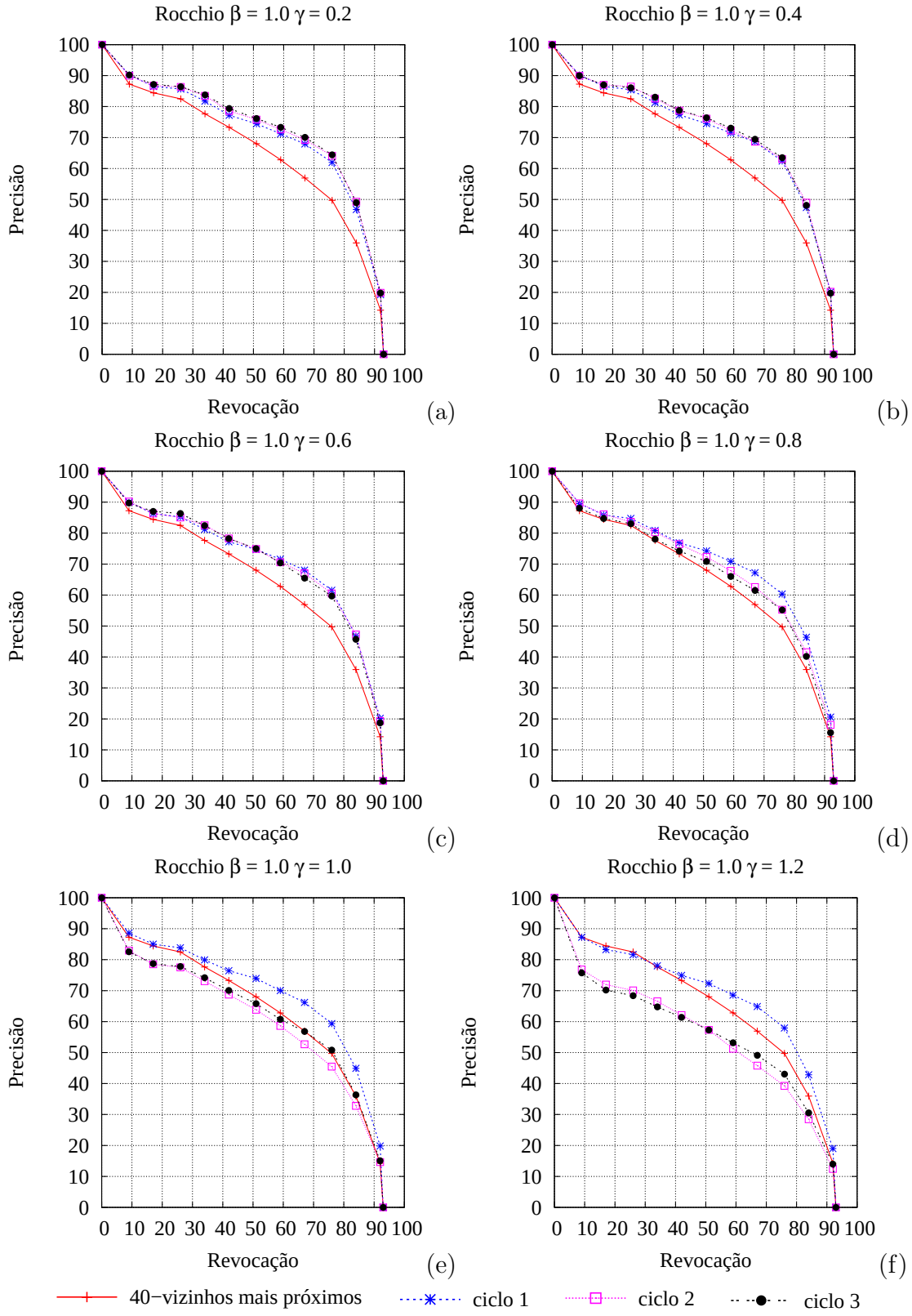

Figura 4.13: Realimentação positiva: gráficos de precisão e revocação para as consultas aos 40-vizinhos mais próximos e os 3 primeiros ciclos de realimentação utilizando a fórmula de Rocchio, com $\beta = 1,0$. (a) $\gamma = 0,2$ (b) $\gamma = 0,4$ (c) $\gamma = 0,6$ (d) $\gamma = 0,8$ (e) $\gamma = 1,0$ (f) $\gamma = 1,2$.

de acesso métrico Slim-tree. Para permitir a comparação em condições equivalentes, o acesso seqüencial foi implementado como um arquivo composto de páginas de tamanho fixo com o mesmo tamanho em *bytes* das páginas das estruturas R*-tree e Slim-tree. Todos os métodos de acesso foram implementados no mesmo arcabouço para permitir uma comparação justa, por exemplo, garantindo que os mesmos tipos de dados são usados na estrutura que representa os elementos de um conjunto de dados para todos os métodos. Os gráficos a seguir apresentam os resultados para execução de 100 consultas por similaridade agregada usando conjuntos de centros de consulta Q distintos compostos por objetos selecionados aleatoriamente. Os experimentos foram executados utilizando a função de distância euclidiana (L_2), em um micro computador Intel Pentium D, de 3,4 GHz, com 1 GB de memória principal e 200 GB de disco rígido. Os gráficos apresentam o comportamento dos algoritmos para os três principais parâmetros normalmente empregados com relação à análise de custo de consultas por similaridade: o número médio de cálculo de distâncias, o número médio de acessos a páginas de disco e o tempo total para executar as consultas.

4.3.2.1 Consultas por Abrangência Agregada

O objetivo do experimento apresentado nesta seção é avaliar o comportamento das consultas por abrangência agregada com relação ao raio agregado ξ, comparando a estratégia proposta MASS com a estratégia Falcon e com o acesso seqüencial. A Figura 4.14 apresenta os resultados das consultas realizadas com o conjunto *Corel Image Features*, composto por 68.040 histogramas de cor (32 dimensões) disponível em [Asuncion e Newman, 2007], considerando $g = 1$, $|Q| = 10$, e variando o raio agregado ξ de 0,5 até 3,5. É importante notar que o algoritmo Falcon baseia-se na união de consultas por abrangência de único centro com raio ϵ estimado pelo usuário. Desse modo, ele somente garante resultados exatos quando o usuário não subestima o valor de ϵ. Para este experimento, o valor de ϵ utilizado foi calculado como a média dos raios das consultas de único centro necessárias para encontrar as respostas exatas computadas por meio de consultas por abrangência agregada no acesso seqüencial, que corresponde ao menor valor seguro para encontrar resultados exatos.

A Figura 4.14-d apresenta o número médio de elementos retornados para cada valor de raio agregado, que resultou em número de elementos com crescimento exponencial. Com raio agregado $\xi = 3,5$, 4,4% do conjunto de dados é retornado (3.000 de um total de 68.040 elementos). Mesmo com essa grande quantidade de dados recuperados, a técnica MASS é 3 vezes mais eficiente que o acesso seqüencial. Dado que o algoritmo Falcon baseia-se na união de consultas por abrangência (que resulta em número de comparações e tempo de ordem quadrática), ele requer maior número de acessos a disco e tempo gasto para raios agregados maiores. Além disso, com o uso do parâmetro ϵ do algoritmo Falcon conforme

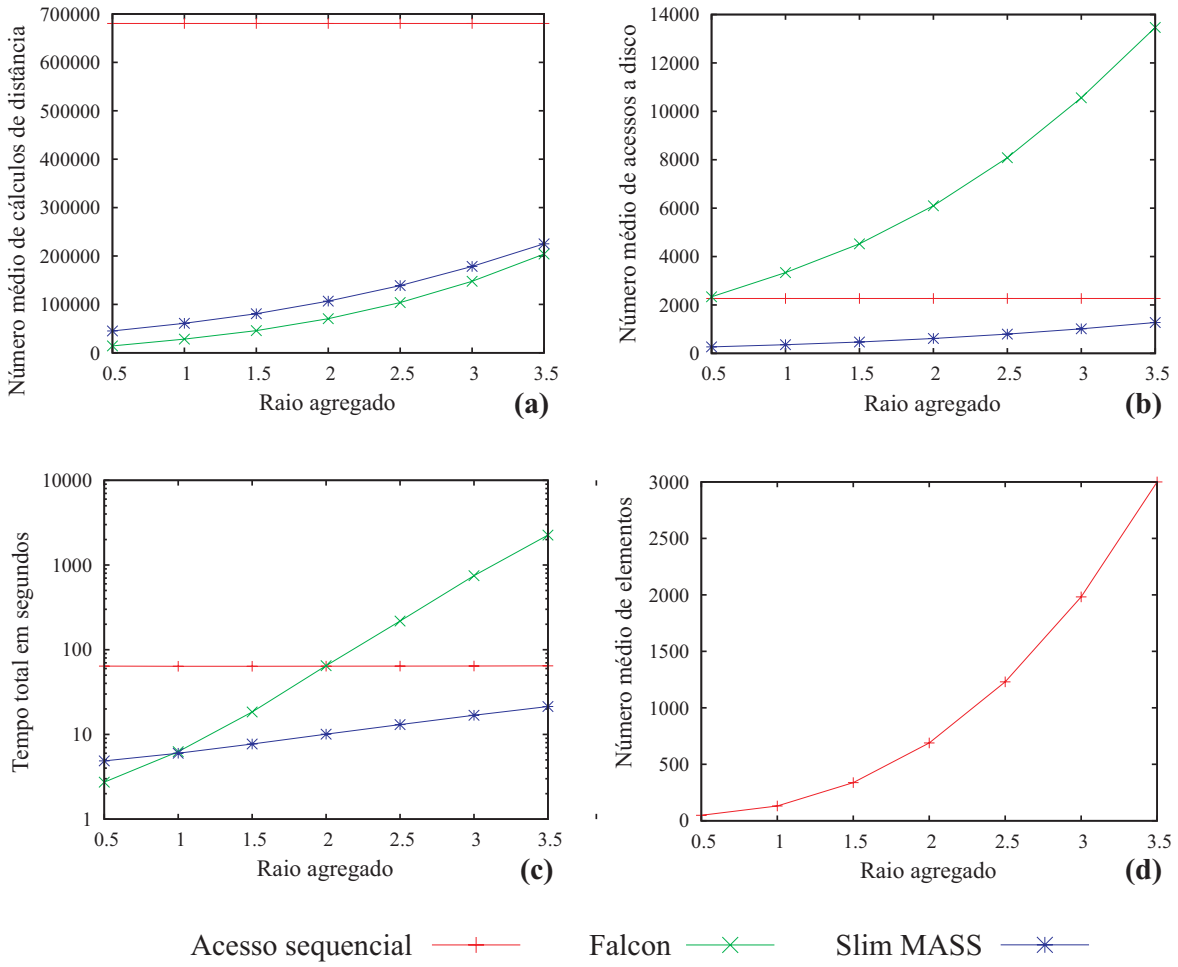

Figura 4.14: Conjunto *Corel Image Features*, consultas por abrangência agregada, $g = 1$, $|Q| = 10$, variação do raio agregado ξ. (a) Número médio de cálculos de distância. (b) Número médio de acessos a disco. (c) Tempo de execução total para 100 consultas em escala logarítmica. (d) Número médio de elementos retornados.

descrição acima, o mesmo resultou em apenas 83% de resultados exatos, ressaltando a dificuldade em prever o valor ϵ que deve ser definido pelo usuário.

4.3.2.2 Consultas aos k-Vizinhos Mais Próximos Agregado

O objetivo do experimento apresentado nesta seção é avaliar o comportamento das consultas aos k-vizinhos mais próximos agregado conforme o número k de elementos aumenta, comparando a estratégia MASS proposta com a estratégia R*-tree MBM e com o acesso seqüencial. Os valores de k variaram de 50 a 500 elementos. Foi empregado o conjunto de histogramas de níveis de cinza (256 dimensões) extraídos do conjunto de imagens denominado *ALOI Object Viewpoint* [Geusebroek et al., 2005], composto por 72.000 imagens. A Figura 4.15 apresenta os resultados para esse conjunto, baseado no fator de agregação $g = 1$ e $|Q| = 15$. O desempenho do método R*-tree MBM foi degenerado devido a natureza do método de acesso R*-tree, que apresenta bom desempenho para dados imersos em

espaços de baixa dimensionalidade. Ainda assim a estratégia MBM resultou em menor número de cálculos de distância e menor tempo para executar se comparada com o acesso seqüencial, mas acessou um número superior de páginas de disco, mesmo para valores pequenos de k. O método MASS manteve o desempenho linear .

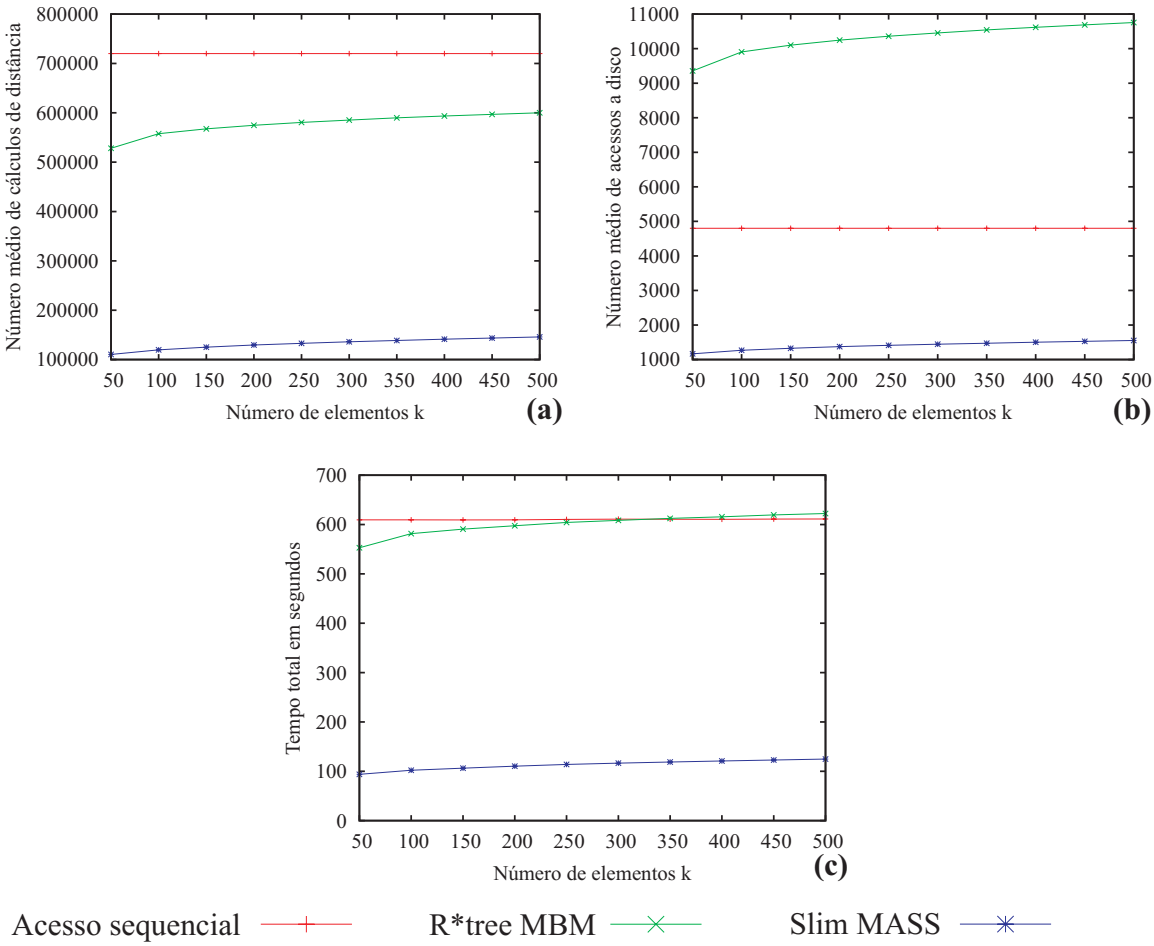

Figura 4.15: Consultas aos k-vizinhos mais próximos agregado sobre o conjunto *ALOI Object Viewpoint*, $g = 1$, $|Q| = 15$, variação do número de elementos k. (a) Número médio de cálculos de distância. (b) Número médio de acessos a disco. (c) Tempo de execução total para 100 consultas.

Considerando a recuperação de $k = 500$ (0,7% do conjunto), que é um valor alto para uma consulta por similaridade, o método MASS computou 4,9 vezes menos cálculos de distância e 3,1 vezes menos acessos a disco, sendo 4,9 vezes mais rápido que o acesso seqüencial. Em geral, uma consulta por similaridade típica submete um valor k de elementos muito menor que o considerado neste experimento, e nesse caso os resultados obtidos pelo método MASS são ainda melhores.

O experimento seguinte avaliou o comportamento das consultas aos k-vizinhos mais próximos agregado conforme o número de centros de consulta e o número k de elementos aumentam. A Figura 4.16 apresenta o resultado para k variando entre 10 e 100 sobre o conjunto *ALOI Object Viewpoint* enquanto a variação do número de centros de consulta

$|Q|$ variando de 2 a 10 centros, apenas para o método MASS. Um efeito interessante pode ser notado neste experimento: enquanto o número de cálculos de distância e o tempo total necessário para executar as consultas aumentam linearmente com relação ao número de centros de consulta $|Q|$ (Figuras 4.16-a e 4.16-c), o incremento do número de acessos a disco diminui logaritmicamente com relação ao incremento do número de centros de consulta $|Q|$ (Figura 4.16-b). Isto ocorre uma vez que, ao aumentar o número de centros de consulta $|Q|$, a região no espaço onde reside o resultado da consulta torna-se mais "esférico", aumentando a capacidade de poda do método e reduzindo o número de páginas candidatas a conter respostas. O tempo total com mesmo comportamento em relação ao número de cálculos de distância, e não ao número de acessos a disco, indica que os cálculos de distância consomem mais tempo que os acessos a disco para o processamento das consultas realizadas neste experimento.

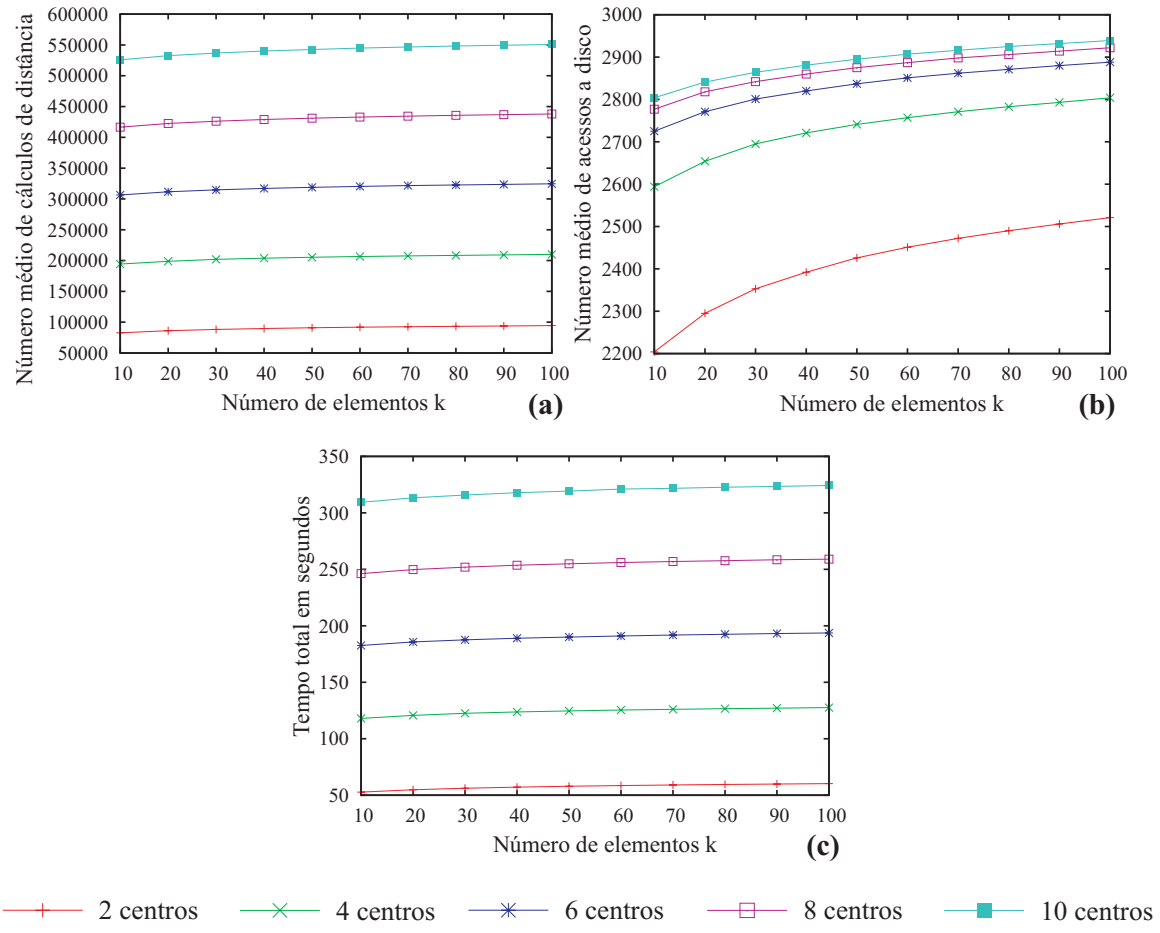

Figura 4.16: Consultas aos k-vizinhos mais próximos agregado sobre o conjunto *ALOI Object Viewpoint*, fator de agregação $g = 2$, variação do número de elementos k e do número de centros de consulta Q. Todas as curvas correspondem à execução do algoritmo MASS. (a) Número médio de cálculos de distância. (b) Número médio de acessos a disco. (c) Tempo de execução total para 100 consultas.

4.3.2.3 Número de Dimensões

O objetivo do experimento apresentado nesta seção é avaliar a escalabilidade da técnica MASS quando aplicada em dados com alta dimensionalidade. Os conjuntos de dados empregados são baseados no ALOI Object Viewpoint [Geusebroek et al., 2005]. Foram gerados 6 conjuntos por meio da extração de histogramas de cor quantizados em 8, 16, 32, 64, 128 e 256 cores para cada uma das 72.000 imagens do conjunto, resultando em 8, 16, 32, 64, 128 e 256 dimensões respectivamente. Como o tamanho em *bytes* dos vetores de características resultantes é diferente para cada conjunto quantizado em um número de cores, os conjuntos foram indexados no método de acesso Slim-tree com página de disco de tamanho proporcional ao tamanho do vetor de características resultante. A Tabela 4.2 apresenta as características dos índices criados.

Tabela 4.2: Índices criados para o experimento de escalabilidade do número de dimensões.

Descrição	Número de dimensões					
	8	16	32	64	128	256
Tamanho da página (em quilobytes)	1	2	4	8	16	32
Tamanho do elemento (em bytes)	36	68	132	260	516	1.028
Elementos / página	28,4	30,1	31,0	31,5	31,8	31,9
Número de páginas do acesso seqüencial	2.880	2.572	2.400	2.323	2.323	2.323
Número de páginas da Slim-tree	5.943	4.925	4.278	4.019	3.824	3.806
Altura da Slim-tree	5	5	4	4	4	4

Os experimentos mediram o número médio de cálculos de distância, acessos a disco e o tempo total para executar 100 consultas aos 20-vizinhos mais próximos com conjuntos de centros sorteados aleatoriamente com cardinalidade $|Q| = 10$ e fator de agregação $g = 0.5$. Os mesmos conjuntos de Q foram usados para realizar as consultas nos conjuntos de 8, 16, 32, 64, 128 e 256 dimensões. A Figura 4.17 apresenta os resultados.

O número de cálculos de distância e de acessos a disco da técnica MASS apresentou comportamento constante com relação ao aumento da dimensionalidade. O número de acessos a disco para o acesso seqüencial diminui com o aumento da dimensionalidade uma vez que páginas de disco maiores permitem acomodar mais elementos em cada página, diminuindo a fragmentação dos dados (as razões entre o número de elementos por página é apresentada na Tabela 4.2). O tempo total requerido para executar as consultas segue um crescimento linear tanto para a técnica MASS quanto para o acesso seqüencial. Entretanto, a técnica MASS aumenta em ritmo menor, como pode ser verificado na Figura 4.17-c.

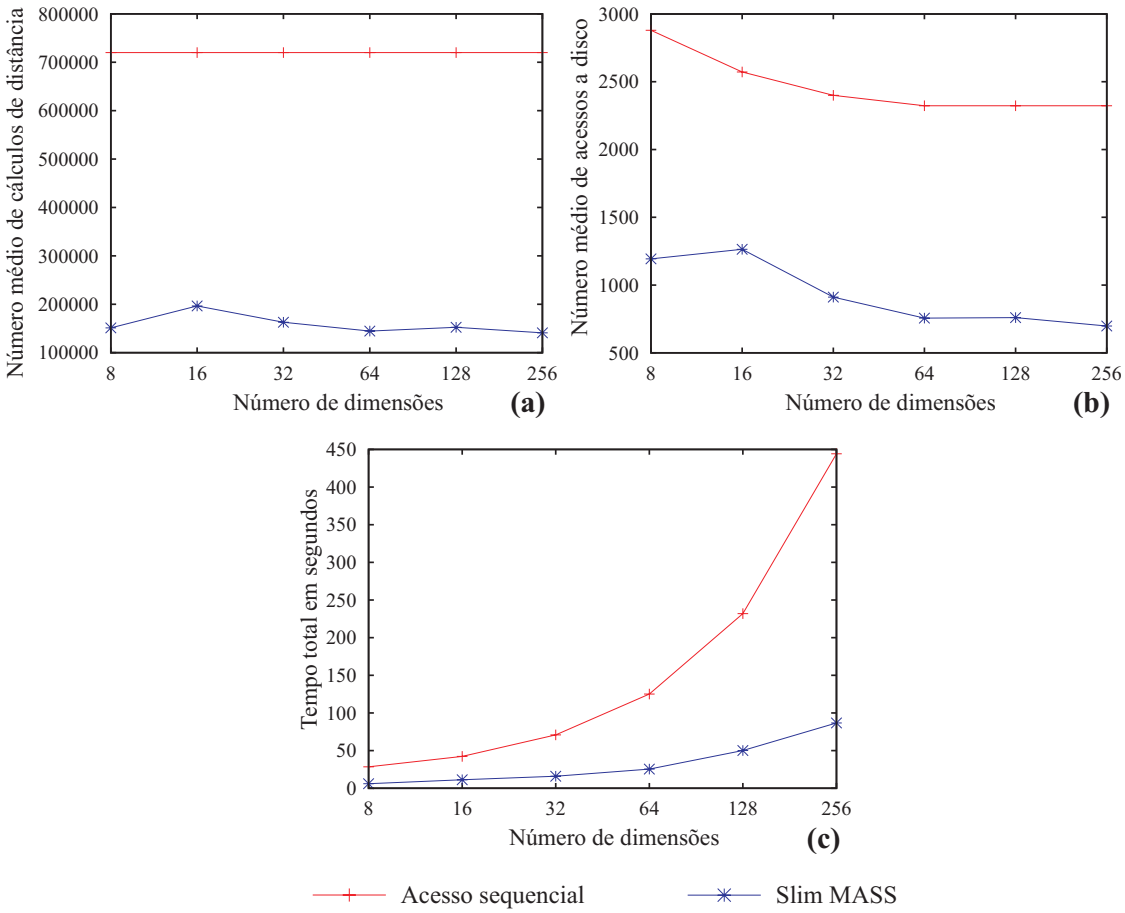

Figura 4.17: Consultas aos k-vizinhos mais próximos agregado sobre conjunto *ALOI Illumination Color*, $g = -\infty$, $|Q| = 10$, $k = 20$. (a) Número médio de cálculos de distância. (b) Número médio de acessos a disco. (c) Tempo de execução total.

4.3.2.4 Número de Elementos

O objetivo do experimento apresentado nesta seção é avaliar a escalabilidade dos métodos com relação ao número de elementos do conjunto para consultas aos k-vizinhos mais próximos agregado, comparando a estratégia proposta com a estratégia R*-tree MBM e com o acesso seqüencial. Para isso foram utilizados conjuntos de dados sintéticos, gerados especialmente para avaliar o comportamento dos algoritmos frente à conjuntos de dados de diferentes tamanhos. Os conjuntos de dados utilizados são baseados em elementos com distribuição gaussiana e agrupados em 10 *clusters* como descrito em [Ciaccia et al., 1997]. Nesse sentido, foram gerados 5 conjuntos de 32 dimensões com 100.000 a 500.000 elementos, os quais foram indexados com Slim-trees e R*-trees com páginas de disco de 4.096 *bytes*. A Figura 4.18 apresenta o comportamento linear dos algoritmos com relação à cálculos de distância, acessos à páginas de disco e tempo total. As curvas referentes ao acesso seqüencial apresentam a proporcionalidade com relação ao número de elementos do conjunto de dados, ou seja $O(n)$. Os métodos R-tree MBM e Slim-tree MASS permitem a realização de podas nas estruturas durante as consultas, e resultam em curvas com

redução dos cálculos de distância, acessos à páginas de disco e tempo total. É importante notar o ganho de desempenho do método Slim-tree MASS em relação ao método R-tree MBM.

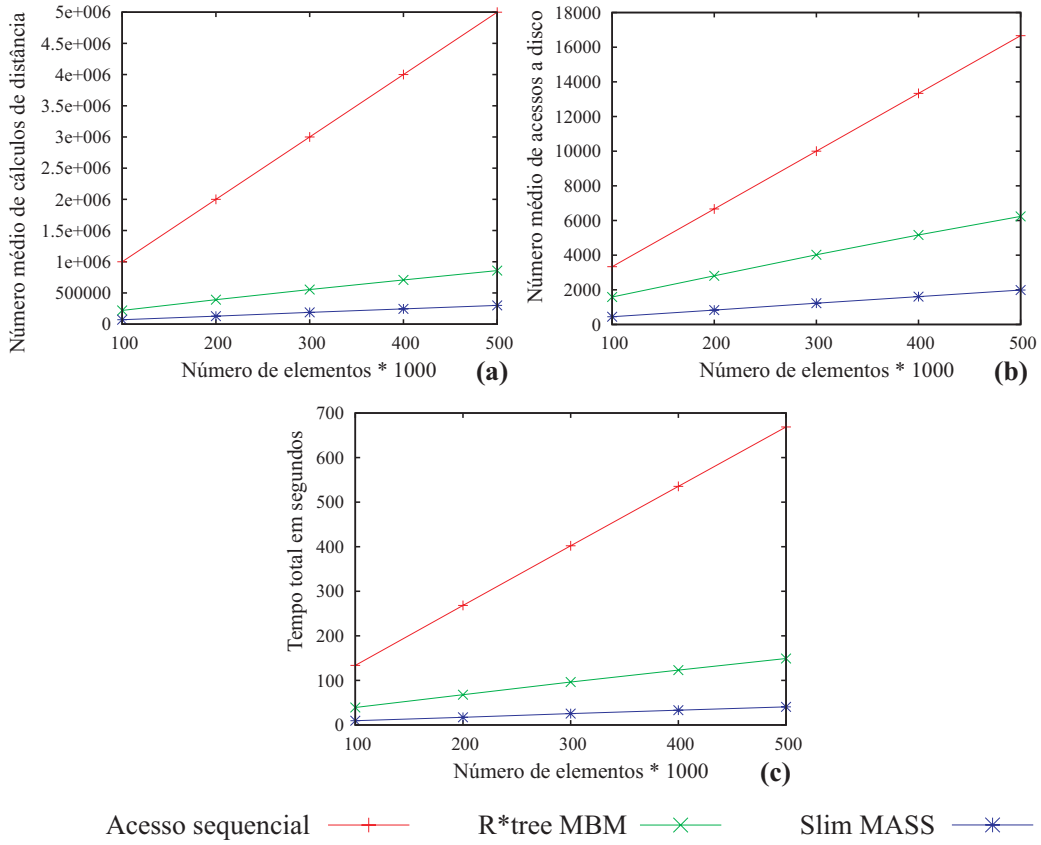

Figura 4.18: Consultas aos k-vizinhos mais próximos agregado sobre conjuntos sintéticos, $g = -\infty$, $|Q| = 10$, $k = 20$. (a) Número médio de cálculos de distância. (b) Número médio de acessos a disco. (c) Tempo de execução total.

A Tabela 4.3 descreve os índices criados para este experimento, que apresentam comportamento linear em relação ao número de páginas de disco e na altura da hierarquia gerada. Apesar do número de páginas de disco das estruturas tipo Slim-tree e R*-tree resultarem em um número de páginas de disco da ordem de duas vezes o número de páginas de disco do acesso seqüencial, verifica-se nas curvas apresentadas na Figura 4.18-b que o número de acessos a disco em consultas é bastante reduzido.

Tabela 4.3: Índices criados considerando o número de elementos dos conjuntos.

Descrição	Número de elementos vezes 1.000				
	100	200	300	400	500
Número de páginas do acesso seqüencial	3.334	6.667	10.000	13.334	16.667
Número de páginas da R*-tree	7.435	14.697	22.226	29.575	36.826
Altura da R*-tree	5	6	6	6	6
Número de páginas da Slim-tree	6.665	13.684	20.782	27.928	35.186
Altura da Slim-tree	5	5	5	5	5

4.4 Considerações Finais

Este capítulo apresentou as consultas por similaridade agregada em espaços métricos, que podem ser consideradas como uma generalização das consultas por similaridade de único centro (abrangência e vizinhos mais próximos) para múltiplos centros de consulta, bem como sua aplicação em métodos de realimentação de relevância em consultas por conteúdo de imagens. Foram apresentadas propriedades importantes dessas consultas, como a propriedade do raio agregado mínimo e o uso de pesos. Os experimentos apresentaram resultados em que a precisão das consultas foi melhorada com o emprego das consultas aos k-vizinhos mais próximos agregados em ciclos de realimentação de relevância. Para comparação, os mesmos experimentos foram executados com a fórmula de Rocchio, nos quais não se conseguiu a qualidade apresentada pelas consultas aos k-vizinhos mais próximos agregados.

Além da eficácia da utilização dessas consultas para realimentação de relevância, outra questão importante diz respeito à sua otimização utilizando os métodos de acesso existentes para consultas por similaridade em espaços métricos. O algoritmo MASS (*Metric Aggregate Similarity Search*) apresentado nesse capítulo é o primeiro método capaz de executar eficientemente consultas por similaridade agregada em dados imersos em espaços métricos, com base na propriedade de desigualdade triangular que permite calcular o limite mínimo de uma região de cobertura para um grande número de funções de agregação. É importante notar que ele sempre fornece respostas exatas às consultas e não depende de parâmetro definido pelo usuário que não sejam os parâmetros da própria definição da consulta, ao contrário do método Falcon. Foram apresentados novos algoritmos para executar consultas por similaridade agregada limitadas por raio ou por número de elementos. Os resultados dos experimentos realizados com conjuntos de dados sintéticos e reais permitiram comparar o desempenho dos algoritmos com os métodos existentes na literatura, mostrando que o método MASS é escalável no número de elementos do conjunto de dados, e se o conjunto estiver imerso no domínio espacial, também no número de dimensões. Além disso, o método MASS demonstrou melhor desempenho que as técnicas Falcon e R*-tree MBM para consultas por similaridade agregadas típicas. Com os bons resultados apresentados, conclui-se que o método MASS pode ser usado para acelerar o processamento de métodos de realimentação de relevância em sistemas de recuperação de imagens por conteúdo.

Capítulo

5

Diversidade em Consultas aos k-Vizinhos mais Próximos

"The most universal quality is diversity"
- Montaigne.

Nas últimas décadas houve um grande interesse na otimização das consultas aos k-vizinhos mais próximos (*k-nearest neighbor queries*). Uma consulta típica aos k-vizinhos mais próximos recebe como parâmetros um ou mais elementos de consulta e um valor k de elementos desejados de um conjunto de dados, e o objetivo é retornar como resposta os k elementos que possuem as menores distâncias ao elemento de consulta, computadas por uma função de distância (como descrito na Seção 4.1). No entanto, essas consultas não consideram o relacionamento desses elementos entre si.

Embora a existência de elementos idênticos em grandes coleções de imagens seja rara, a existência de elementos muito similares é comum, sendo que muitas vezes esses elementos são pertencentes ao mesmo grupo de imagens, por exemplo, imagens de um mesmo exame de tomografia computadorizada, fotografias de um mesmo objeto em diferentes momentos ou imagens de uma mesma página na *web*. No caso de coleções de imagens de exames de tomografia computadorizada, ao procurar pelas 10 imagens mais próximas de uma imagem de referência, o usuário provavelmente não estará interessado em imagens do mesmo exame ou do mesmo paciente, mas nas imagens mais próximas e que sejam de outros 10 pacientes. Em suma, durante a fase exploratória de uma consulta complexa, o usuário pode não interessar-se exatamente pelas imagens mais próximas, mas nas imagens mais próximas que atendam a um critério de diversidade. Nesse sentido, a diversidade

entre os elementos de uma consulta por vizinhos mais próximos pode gerar resultados com melhor qualidade semântica, reduzindo a descontinuidade existente nas consultas por conteúdo de imagens.

Existem várias aplicações que podem se beneficiar da diversidade do conjunto resposta. Por exemplo, a pesquisa pelo termo "*apple*" em uma máquina de busca da *web*, seria mais interessante que, ao invés da consulta inicial exploratória retornar os primeiros 100 resultados relacionados apenas ao fabricante de microcomputadores, que ela retornasse alguns resultados referentes à computadores, outros referentes à cantora "Fiona Apple", outros referentes à cidade conhecida como "*big apple*", outros referentes à fruta, e etc. Isso motivou o desenvolvimento do trabalho descrito neste capítulo, que considera o problema de prover diversidade nos resultados de consultas aos k-vizinhos mais próximos, aqui denominado k-vizinhos diversos mais próximos. A motivação consiste em recuperar um conjunto de elementos próximos ao elemento de consulta que também satisfaça a restrição de apresentar diversidade de características entre seus elementos. O processamento da consulta aos k-vizinhos diversos mais próximos deve empregar uma função de ordenação que considere a similaridade ao elemento de referência e também a diversidade entre seus elementos – ambas computadas por funções de distância. A Figura 5.1 apresenta a intuição da consulta em um espaço bidimensional com distância L_1. Em uma consulta aos 2 vizinhos mais próximos ao elemento s_q, $\{A, B\}$ formam naturalmente a resposta da consulta, entretanto $\{A, C\}$ são elementos muito próximos do elemento de consulta mas são mais distantes entre si quando comparados com $\{A, B\}$.

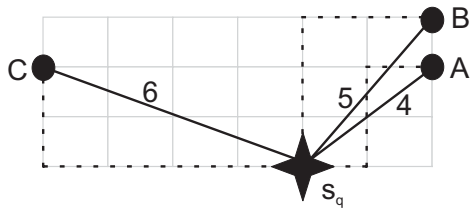

Figura 5.1: Como capturar a diversidade em uma consulta aos k-vizinhos mais próximos?

Este capítulo apresenta uma abordagem para a seleção dos k-vizinhos diversos mais próximos, computada de modo eficiente. O método auxilia na obtenção de resultados com maior semântica. A Seção 5.1 formaliza esse tipo de consulta. A Seção 5.2 apresenta os algoritmos propostos e a Seção 5.3 apresenta os experimentos realizados, seguido da Seção 5.4 que apresenta as considerações finais.

5.1 Consulta aos k-Vizinhos Diversos Mais Próximos

O uso crescente de dados multimídia aumenta os desafios para organizá-los e recuperá-los eficientemente. O objetivo é ser capaz de aumentar a relevância dos resultados de

consultas aos k-vizinhos mais próximos permitindo que o usuário especifique o quão diverso deseja os resultados. Uma vez que as principais consultas realizadas sobre esses dados levam em consideração a dissimilaridade, os espaços métricos são adequados para representá-los, de modo que apenas os elementos e as distâncias entre pares de elementos são considerados.

Uma consulta aos k-vizinhos diversos mais próximos deve encontrar um conjunto de k elementos que minimize sua distância ao centro de consulta s_q e também maximize sua diversidade, ou seja, a distância entre os elementos da resposta. Conforme descrito na Seção 2.2, um espaço métrico envolve um domínio de dados $\mathbb{S}$ e uma função de distância δ. A consulta aos k-vizinhos diversos mais próximos envolve duas funções de distância. Embora seja possível que a mesma função seja usada, definir a consulta baseada em duas funções a torna mais flexível e mais voltada a situações reais. Assim, de um ponto de vista formal, essa consulta é definida sobre um domínio de dados $\mathbb{S}$ que é compartilhado por dois espaços métricos $\mathcal{M}_{sim} = \langle \mathbb{S}, \delta_{sim}() \rangle$ e $\mathcal{M}_{div} = \langle \mathbb{S}, \delta_{div}() \rangle$.

Dado um domínio de dados $\mathbb{S}$ e um conjunto $S \subseteq \mathbb{S}$, o objetivo de uma consulta aos k-vizinhos diversos mais próximos é encontrar um conjunto de k elementos em S que minimize suas distâncias δ_{sim} em relação ao elemento de consulta $s_q \in \mathbb{S}$ (Definição 6) e também maximize sua diversidade (Definição 7), medida pela soma das distâncias δ_{div} entre todos os pares de elementos do conjunto resposta R. As funções de distância δ_{sim} e δ_{div} podem ser funções distintas, e podem ser aplicadas a conjuntos distintos de atributos de cada elemento de um conjunto de dados.

Para definir a consulta aos k-vizinhos diversos mais próximos, definimos primeiramente a medida de similaridade de um conjunto de elementos R a um dado centro de consulta s_q e a medida de diversidade de R. O conjunto $R \subseteq S$ representa um conjunto de elementos candidato a resposta da consulta.

Definição 6. Medida de similaridade. Dado um elemento de consulta $s_q \in \mathbb{S}$ e um conjunto de k elementos $R \subseteq S$, a medida de similaridade $sim : \mathbb{S} \times \mathbb{S}^k \rightarrow \mathbb{R}^+$ de R com relação à s_q é dada por:

$$sim(s_q, R) = \sum_{i=1}^{k} \delta_{sim}(s_q, R_i)$$

Definição 7. Medida de diversidade. Dado um conjunto $R \subseteq S$, a medida de diversidade $div : \mathbb{S}^k \rightarrow \mathbb{R}^+$ de R é dada por:

$$div(R) = \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \delta_{div}(R_i, R_j)$$

É possível normalizar essas equações, de modo que a equação apresentada na Definição 6 pode ser normalizada pelo número de elementos do conjunto resposta (k) e a equação

apresentada na Definição 7 pode ser normalizada pelo número de distâncias entre todos os pares de elementos do conjunto resposta $(k * (k-1)/2)$.

O problema se resume a encontrar o conjunto de elementos que minimize a função de pontuação apresentada na Definição 8. O objetivo é ter um conjunto resultado diverso, que balanceie a relação entre a maximização da diversidade e a minimização das distâncias em relação ao elemento de consulta s_q. O compromisso entre a diversidade e a similaridade é dado pelo peso da diversidade w_{div}, de modo que $0 \leq w_{div} \leq 1$, o que garante flexibilidade ao usuário para determinar a prioridade da similaridade ou da diversidade. Assim, a consulta aos k-vizinhos diversos mais próximos com elemento de consulta s_q sobre um conjunto de dados $S \subseteq \mathbb{S}$ é definida a seguir.

Definição 8. Consulta aos k-vizinhos diversos mais próximos (k-*nearest diverse neighbor query* − k-*NDNq*). Seja s_q um elemento de consulta e $S \subseteq \mathbb{S}$ um conjunto de dados. A consulta aos k-vizinhos diversos mais próximos é o conjunto $R \subseteq S, |R| = k$ que minimiza a função de pontuação : $\mathbb{S} \times \mathbb{S}^k \rightarrow \mathbb{R}^+$ com relação ao elemento s_q.

$$\text{pontuação}(s_q, R) = (1 - w_{div}) \cdot sim(s_q, R) - w_{div} \cdot div(R)$$

No caso de $w_{div} = 0$, o problema equivale à consulta tradicional aos k-vizinhos mais próximos. Por outro lado, quando $w_{div} = 1$, o problema equivale ao problema da diversidade máxima descrito na Seção 3.5.1, que consiste em identificar o conjunto de k elementos mais diversos em um conjunto de dados.

Considere o exemplo apresentado na Figura 5.1 e a consulta aos 2-vizinhos diversos mais próximos considerando a função de distância L_1. Para $w_{div} = 0$, o conjunto resposta corresponde aos elementos $\{A, B\}$. Entretanto, para $w_{div} = 0,2$:

$$\begin{aligned}
\text{pontuação}(s_q, \{A, B\}) &= 0,8 \cdot (4 + 5) - 0,2 \cdot 1 = 7,0 \\
\text{pontuação}(s_q, \{A, C\}) &= 0,8 \cdot (4 + 6) - 0,2 \cdot 6 = 6,8 \\
\text{pontuação}(s_q, \{B, C\}) &= 0,8 \cdot (5 + 6) - 0,2 \cdot 7 = 7,4
\end{aligned}$$

logo a minimização da função de pontuação resulta nos elementos $\{A, C\}$ como os 2-vizinhos diversos mais próximos para uma diversidade de 0,2. Nesse exemplo, pela distribuição espacial dos elementos pode-se verificar que a diversidade do elemento C com relação ao elemento A foi levada em consideração e resultou em uma pontuação menor que a do elemento B em relação ao elemento A.

5.1.1 Complexidade

A função de pontuação para um conjunto de k elementos apresenta complexidade de tempo de $O(k^2)$. Dado um conjunto $S \subseteq \mathbb{S}$ com cardinalidade $|S| = n$, há ${}^nC_k$ possíveis

combinações de k elementos. Baseando-se na consulta apresentada na Definição 8, não foi encontrada uma propriedade que permita predizer, para um dado conjunto com mais de k elementos, o valor máximo ou mínimo da função de pontuação. Assim, para encontrar um conjunto de k elementos de um conjunto S que resulte no menor valor para a função de pontuação, há ${}^{n}C_{k} = n!/(k! \cdot (n-k)!)$ combinações de k elementos possíveis para serem avaliadas, resultando na complexidade de tempo de $O(n^k)$, ou seja, uma complexidade de tempo polinomial para qualquer k constante. Assim, é improvável que seja encontrado um algoritmo que garanta uma solução exata em tempo razoável, e o objetivo passa a ser o de encontrar um algoritmo aproximado cuja pontuação seja a menor possível.

O objetivo da consulta aos k-vizinhos diversos mais próximos é aumentar a relevância dos resultados por meio da avaliação da diversidade. Este objetivo fornece pistas de como reduzir o espaço de busca de modo que elementos não interessantes possam ser excluídos do processamento, sem ferir a relevância do resultado da consulta. Um modo de reduzir o espaço de busca consiste em atribuir um valor de distância máximo para δ_{sim} em relação ao elemento de consulta s_q, assumindo que elementos com distância maior que esse valor não são relevantes. Esta restrição permite levar em consideração apenas os ℓ elementos mais próximos à s_q do que essa distância máxima. Outra forma de reduzir o espaço de busca é prover diretamente um valor para ℓ, tanto absolutamente quanto proporcionalmente ao valor de k, selecionando de S os ℓ-vizinhos mais próximos ao elemento de consulta considerando a métrica δ_{sim}. Neste trabalho não foi definido como determinar um valor para ℓ apropriado, dado que um valor de $\ell \geq k$ seja fornecido.

Para um sub-conjunto de S contendo no máximo ℓ elementos mais próximos, é preciso avaliar a função de pontuação para apenas ${}^{\ell}C_{k}$ combinações de k elementos. Sem considerar o problema de como definir um valor apropriado para ℓ, é importante notar que, mesmo para pequenos valores de k e ℓ, o número total de combinações possíveis para minimizar a função de pontuação é grande. Por exemplo, para $k = 10$ e $\ell = 100$, há aproximadamente 10^{13} combinações possíveis para serem avaliadas.

O Algoritmo 6 é uma função recursiva que computa a solução ótima por meio de uma busca exaustiva e é denominado k-NDNq-Exaustivo. A recursão é apresentada no Passo 4. Por questão de desempenho, uma matriz de dissimilaridade $\ell \times \ell$ deve ser pré-computada e empregada pela função de pontuação para evitar a computação de distâncias entre um mesmo par de elementos mais de uma vez.

5.1.2 Limite da Consulta

A dificuldade em criar um algoritmo eficiente para minimizar a função de pontuação se deve ao fato de não existir uma propriedade que permita inferir que a troca de um elemento $s_i \in R$ por qualquer outro elemento aumente ou reduza as medidas de similaridade e

Algoritmo 6 k-NDNq-Exaustivo: função recursiva que computa a função de pontuação de todos os conjuntos possíveis de k elementos de um conjunto de dados

Entrada: k, *índiceResposta*, *posElementoNoConjunto*, vetor *resposta*

```
1: para posiçãoCorrente ← posElementoNoConjunto até |conjunto| faça
2:    resposta[índiceResposta] ← posiçãoCorrente
3:    se índiceResposta+1 < k então
4:       k-NDNq-Exaustivo(k, índiceResposta+1, posiçãoCorrente+1, resposta)
5:    fim se
6:    se posiçãoCorrente+1 < |conjunto| então
7:       computar a função de pontuação como pontuação
8:       se pontuação < melhorPontuação então
9:          atualizar melhorPontuação e manter o vetor resposta
10:      fim se
11:   fim se
12: fim para
```

diversidade. Para se avaliar a possibilidade de um i-ésimo elemento s_i mais próximo de s_q fazer parte da resposta, é necessário computar a função de pontuação para todas as combinações de k elementos (que contenham o elemento s_i) com os $i-1$ elementos já avaliados. Assim, não é possível definir um limite superior para a função de pontuação e portanto não é possível determinar o critério de parada em um algoritmo de busca aos vizinhos mais próximos que garanta que a solução ótima faça parte dos elementos encontrados até a parada.

Uma consulta aos k-vizinhos diversos mais próximos com restrição no espaço de busca resulta no conjunto de k elementos entre os mais próximos limitados por: uma distância máxima ξ em relação ao elemento de consulta s_q; ou um valor ℓ de elementos mais próximos. Em ambos os casos, o resultado corresponde ao conjunto de k elementos que minimiza a função de pontuação apresentada na Definição 8.

Além de ser possível definir uma restrição no espaço de busca, determinando que o usuário não está interessado nos elementos mais distantes que um elemento de posição ℓ em relação ao elemento de consulta ou que um determinado raio ξ, a seguinte heurística pode ser definida para determinar a condição de parada do algoritmo incremental [Roussopoulos et al., 1995] de busca aos k-vizinhos mais próximos. O Algoritmo 7 apresenta o critério de parada para o algoritmo incremental considerando a função de pontuação. A variável *critérioParada* delimita o espaço de busca (Passo 9). A cada iteração, o espaço de busca é incrementado em k elementos e uma solução é selecionada, até que o algoritmo não encontre uma solução melhor que a solução encontrada na iteração anterior. No algoritmo, os passos para a leitura dos nodos do método de acesso e a inclusão em *prioridade* foram resumidos no Passo 6. A variável *prioridade* pode ser implementada como uma árvore binária balanceada de busca ordenada pela distância em relação ao elemento de consulta, na qual a fase incremental do algoritmo insere as sub-árvores e elementos,

permitindo que sejam acessados em ordem crescente de distância. No Passo 8, o próximo elemento mais próximo é inserido no resultado da consulta aos vizinhos mais próximos e, a cada *critérioParada*k* elementos inseridos nesse conjunto armazenado em *knn*, os elementos selecionados são submetidos à heurística desejada (Passo 10) para encontrar uma solução para a consulta.

Algoritmo 7 Consulta aos k-vizinhos mais próximos incremental considerando a função de pontuação como critério de parada.

Entrada: elemento de consulta e, quantidade k de elementos, peso w_{div}

```
 1: prioridade ← ∅
 2: knn ← ∅, melhorResultado ← ∅
 3: melhorResultado.pontuação ← ∞
 4: critérioParada ← 2
 5: enquanto parada = falso e prioridade.próximo() faça
 6:    leitura dos nodos do método de acesso e a inclusão em prioridade
 7:    se prioridade.próximo().tipo = elemento então
 8:       knn.adicione(elemento, distância)
 9:       se knn.numEntradas()  mod  (critérioParada * k) = 0 então
10:          resultado ← Heurística(knn, k)
11:          se resultado.pontuação < melhorResultado.pontuação então
12:             melhorResultado ← resultado
13:             critérioParada ← critérioParada +1
14:          senão
15:             parada ← verdadeiro
16:          fim se
17:       fim se
18:    fim se
19: fim enquanto
20: retornar melhorResultado
```

Com o aumento do espaço de busca, se uma solução melhor existir, poderá ser encontrada pelo algoritmo escolhido no Passo 10, e consequentemente ao menos uma iteração a mais será executada. Apesar de não ser possível determinar se a solução encontrada está próxima da solução ótima, o algoritmo tenta encontrar a melhor solução delimitada pelo espaço de busca.

5.2 Algoritmos Propostos

As definições apresentadas na Seção 5.1.2 permitem a recuperação de um conjunto de elementos que contém uma solução para o problema da consulta aos k-vizinhos diversos mais próximos com restrição no espaço de busca. A avaliação da função de pontuação para ${}^{\ell}C_k$ ao invés de ${}^{n}C_k$ conjuntos de k elementos é uma grande melhoria. Entretanto, ${}^{\ell}C_k$ ainda pode ser uma tarefa que requer muito tempo para sua execução. Assim, entre as

abordagens que podem ser empregadas para para encontrar soluções úteis estão os algoritmos gulosos e as heurísticas baseadas em buscas aleatórias. A seguir são apresentados os algoritmos propostos com base nessas estratégias.

5.2.1 Algoritmo k-NDNq-Guloso

Algoritmos gulosos realizam a escolha que apresenta melhor resultado em cada estágio, baseando-se em uma solução ótima local na esperança que esta escolha levará a uma solução ótima global. Uma solução gulosa aproximada desenvolvida para o problema é apresentada no Algoritmo 8, denominado k-NDNq-Guloso. O algoritmo recebe como entrada o valor k desejado e um conjunto de elementos de cardinalidade ℓ ordenados pela distância em relação ao elemento de consulta. No passo 1 são selecionados os primeiros k-vizinhos mais próximos como a solução inicial. Então, para cada elemento seguinte ordenado pela distância ao elemento de consulta, o algoritmo computa a pontuação considerando a troca com cada um dos k elementos selecionados, e mantém na solução a troca com o elemento que resultou em menor pontuação. A complexidade de tempo do algoritmo k-NDNq-Guloso é de $O(k^3 * \ell)$, incluindo o custo da função de pontuação.

Algoritmo 8 Algoritmo k-NDNq-Guloso.

Entrada: k, *elementos*, w_{div}

1: *solução* ← k-vizinhos mais próximos
2: *soluçãoGlobal* ← *solução*
3: calcular a pontuação da *solução* como *pontuaçãoGlobal*
4: $\ell \leftarrow |elementos|$
5: **para** $i \leftarrow k+1$ até ℓ **faça**
6: *pontuaçãoLocal* ← ∞
7: **para** $j \leftarrow 1$ até k **faça**
8: calcular a *pontuação* de *solução* considerando a troca de j com i
9: **se** *pontuaçãoLocal* > *pontuação* **então**
10: *pontuaçãoLocal* ← *pontuação* e manter *solução*
11: **fim se**
12: **fim para**
13: **se** *pontuaçãoGlobal* > *pontuaçãoLocal* **então**
14: *pontuaçãoGlobal* ← *pontuaçãoLocal*
15: *soluçãoGlobal* ← *solução*
16: **fim se**
17: **fim para**
18: retornar *soluçãoGlobal*

5.2.2 Algoritmo k-NDNq-Grasp

O segundo algoritmo proposto é baseado na meta-heurística Grasp descrita na Seção 3.5.1. A seguir são descritas as estratégias empregadas pelas duas fases da heurística para desenvolver o algoritmo para encontrar soluções para uma consulta k-NDNq baseado em busca aleatória.

5.2.2.1 Fase de Construção

Para construir a lista restrita de candidatos (RCL) utilizada pelo k-NDNq-Grasp é necessário definir a medida de benefício para o problema das consultas aos k-vizinhos diversos mais próximos. A medida de benefício computa a contribuição individual de cada elemento $e_i, e_j \in R$, considerando como alvo a função de pontuação apresentada na Definição 8. A contribuição individual é baseada na distância de cada elemento ao elemento de consulta e na distância para outro candidato a fazer parte da resposta, como definido na Equação 5.1. A medida de benefício é a soma dos k menores valores de contribuição(e_i, e_j), como apresentado na Equação 5.2. Finalmente, a RCL é a lista ordenada de $((n-k)/2)$ elementos que apresentam as menores medidas de benefício.

$$\text{contribuição}(e_i, e_j) = (1 - w_{div}) \cdot \delta_{sim}(s_q, e_i) - w_{div} \cdot \delta_{div}(e_i, e_j) \tag{5.1}$$

$$\text{benefício}(e_i) = \sum_{j=1}^{k} \text{contribuição}(e_i, e_j) \tag{5.2}$$

O próximo passo na fase de construção emprega um filtro aleatório controlado que começa com um conjunto baseado nos elementos com as menores medidas de benefício. O filtro consiste em construir soluções aleatórias a partir da RCL. O Algoritmo 9 apresenta a fase de construção do método k-NDNq-Grasp proposto.

5.2.2.2 Fase de Busca Local

A fase de busca local inicia com a solução obtida pela fase de construção. A vizinhança é definida pelo conjunto de soluções obtidas pela troca de um elemento da solução por outro da RCL que não pertence à solução. Para cada elemento e_i e e_j selecionado aleatoriamente, a melhora dada pela troca do elemento e_i pelo elemento e_j é computada. Se para todo $e_i, e_j \in R$ a pontuação global não melhora, a busca termina. Senão, uma nova solução é criada aleatoriamente e uma nova busca local é realizada. O Algoritmo 10 apresenta a fase de busca local do método k-NDNq-Grasp proposto.

Algoritmo 9 Algoritmo de construção do k-NDNq-Grasp.

Entrada: k, *elementos*, w_{div}

construir a lista restrita de candidatos (RCL)
selecionar os primeiros k elementos da RCL como solução inicial
pontuaçãoDeConstrução ← pontuação da solução inicial
para $i = 1$ até *númeroMáximo Filtros* **faça**
    selecionar aleatoriamente k elementos distintos da RCL
    calcular função de pontuação como *pontuação*
    se *pontuação* < *pontuaçãoDeConstrução* **então**
        pontuaçãoDeConstrução ← *pontuação*
        manter a seleção como *soluçãoDeConstrução*
    fim se
fim para
retornar *soluçãoDeConstrução*

5.2.2.3 Path Relinking – PR

Para a realização da estratégia denominada *path relinking* é necessário computar a lista de soluções elite, conforme descrito na Seção 3.5.1.2. Após cada busca local, se a solução local encontrada for melhor que a pior solução elite e a solução local também for suficientemente diferente de todas as soluções elite, então a solução local corrente deve ser adicionada à lista de soluções elite. O critério empregado para determinar se duas soluções são suficientemente diferentes é baseado em um número empírico de elementos diferentes entre duas soluções em avaliação. Para os experimentos apresentados neste capítulo, o valor considerado é $k/4$.

O método que realiza o *path relinking* pode ser executado a cada iteração do k-NDNq-Grasp. Para cada elemento da lista de soluções elite, o método seleciona a melhor solução encontrada ao longo do caminho para a melhor solução global. Em seguida o método verifica se a solução é melhor que a solução global e se a solução pode ser adicionada à lista de soluções elite. O Algoritmo 11 apresenta o método para a computação da solução por meio do *path relinking* entre duas soluções elite que pode ser aplicado na fase de busca local do método k-NDNq-Grasp proposto.

5.2.2.4 Expansão da Vizinhança – EV

O método de expansão da vizinhança foi desenvolvido como uma estratégia de intensificação da busca local. O objetivo é a expansão do espaço de busca dos elementos da vizinhança da melhor solução encontrada por uma iteração de busca local do k-NDNq-Grasp. A intensificação emprega os vizinhos mais próximos de cada elemento da solução, uma vez que esses elementos podem ser considerados candidatos à geração de soluções com alta qualidade. Como na estratégia *path relinking*, esses elementos são empregados

Algoritmo 10 Algoritmo de busca local do k-NDNq-Grasp.

Entrada: k, *elementos*, w_{div}, *soluçãoDeConstrução*

```
1: pontuaçãoGlobal ← ∞
2: para m ← 1 até númeroDeBuscasLocais faça
3:    pontuaçãoLocal ← ∞
4:    n ← 1
5:    enquanto n < númeroDeTrocas faça
6:       substituir aleatoriamente um elemento e_i da solução com um elemento e_j da RCL
7:       computar a função de pontuação como pontuação
8:       se pontuação < pontuaçãoLocal então
9:          pontuaçãoLocal ← pontuação
10:         n ← 1
11:      senão
12:         n ← n + 1 e desfaça substituição
13:      fim se
14:   fim enquanto
15:   computar a função de pontuação como pontuação
16:   se pontuação < pontuaçãoGlobal então
17:      pontuaçãoGlobal ← pontuação
18:      manter a seleção como soluçãoGlobal
19:   fim se
20: fim para
21: retornar soluçãoGlobal
```

em operações de substituição com um elemento da solução de cada vez, por um número limitado de iterações. O Algoritmo 12 apresenta o método para a intensificação da fase de busca local do método k-NDNq-Grasp proposto.

5.3 Experimentos

Os experimentos foram executados em um único núcleo de um microcomputador equipado com processador Intel Core 2 Duo (modelo 6320) de 1,86 GHz. A apresentação dos resultados é dada pela medida *gap* das médias das pontuações obtidas acompanhadas dos respectivos tempos de execução e pelos perfis de desempenho (*performance profiles*), apresentados na Seção 3.6. Um gráfico de perfis de desempenho permite avaliar a probabilidade de um algoritmo obter desempenho melhor que os outros algoritmos em análise.

Os experimentos foram realizados com conjuntos de dados disponíveis na biblioteca SISAP [Figueroa et al., 2009] e no banco de dados bibliográficos DBLP [Ley, 2009], e são descritos na Tabela 5.1. Para os conjuntos DBLP e Documentos, a dimensionalidade representa o número total de termos distintos entre todos os elementos, e a função de distância empregada é o inverso da similaridade do cosseno. Os vetores que representam

Algoritmo 11 Algoritmo *path relinking* do k-NDNq-Grasp-PR.

Entrada: *solução1*, *solução2*, k, w_{div}

```
1: contar quantos elementos fazem parte de ambas solução1 e solução2
2: melhorPontuação ← pontuação(solução2)
3: para m ← 1 até númeroDeElementosDiferentes faça
4:    substituir o m-ésimo elemento diferente da solução1 pelo elemento da solução2
5:    pontuação ← pontuação(solução1)
6:    se pontuação < melhorPontuação então
7:       melhorPontuação ← pontuação
8:       manter solução1 como solução
9:    fim se
10: fim para
11: retornar solução
```

Algoritmo 12 Algoritmo expansão da vizinhança do k-NDNq-Grasp-EV.

Entrada: *solução*, k, *elementos*, w_{div}

```
1: para i ← 1 até k faça
2:    encontrar os p-vizinhos mais próximos de cada elemento i da solução
3: fim para
4: para m ← 1 até númeroDeTrocas faça
5:    substituir aleatoriamente um elemento i da solução com um elemento j do conjunto
      de vizinhos mais próximos do elemento i
6:    computar a função de pontuação da nova solução como pontuação
7:    se pontuação < pontuaçãoSolução então
8:       pontuaçãoSolução ← pontuação
9:    senão
10:      desfaça substituição
11:   fim se
12: fim para
13: retornar solução
```

esses elementos são esparsos, variando entre 9 a 15.116 para o conjunto Documentos e de 1 to a 68 para o conjunto DBLP. A representação desses vetores é dada por pares $\{\text{posição}, \text{valor}\}$, sendo que o valor de cada dimensão é resultado da freqüência do termo normalizada no intervalo $[0, 1]$. O conjunto DBLP foi desnormalizado da seguinte maneira: como cada publicação contém até m autores, foram criadas m tuplas para cada publicação, composta por autor, título e ano de publicação. O conjunto resultante contém 3 milhões de tuplas (1 milhão de publicações e média de 3 autores por publicação), e o vetor de freqüência de termos contém os termos dos títulos. Os valores apresentados nos experimentos a seguir são resultados da média de 20 consultas aleatórias.

Os resultados da avaliação empírica dos métodos propostos são comparados com os resultados do algoritmo exaustivo que computa a solução ótima considerando o espaço de busca da solução restrito aos 200-vizinhos mais próximos e considerando espaços de busca maiores sem a comparação com o algoritmo exaustivo. Os métodos avaliados são

Tabela 5.1: Conjuntos de dados empregados nos experimentos.

Conjunto de dados	Número de elementos	Número de dimensões	Função de distância	Fonte
Nasa	40.150	20	Manhattan	[Figueroa et al., 2009]
Cores	112.544	112	Euclidiana	[Figueroa et al., 2009]
Faces	1.016	761	Euclidiana	[Figueroa et al., 2009]
Documentos	25.276	237.781	Cosseno^{-1}	[Figueroa et al., 2009]
DBLP	3.000.000	148.732	Cosseno^{-1}	[Ley, 2009]

denominados nos gráficos: k-NDNq-Exaustivo, k-NDNq-Guloso, e k-NDNq-Grasp, que foram avaliados considerando as estratégias de otimização *path relinking* (k-NDNq-Grasp-PR) e expansão da vizinhança (k-NDNq-Grasp-EV). São reportados os gráficos de perfis de desempenho e as tabelas com as medidas de *gap* entre os métodos.

5.3.1 Exemplo de Consulta

Para exemplificar a utilidade da consulta aos k-vizinhos diversos mais próximos, considere o seguinte exemplo, baseado no conjunto DBLP. A consulta centrada nos termos *"Nearest Neighbor Algorithms"* considerando os 20-vizinhos mais próximos resultou nas tuplas apresentadas na Tabela 5.2.

Tabela 5.2: Consulta aos 20-vizinhos mais próximos tradicional. Elemento de consulta: "Nearest Neighbor Algorithms"

Elemento	δ
X. Zhang, Kernel Nearest Neighbor Algorithm. 2002	0.523
L. Ji, Kernel Nearest Neighbor Algorithm. 2002	0.523
K. Yu, Kernel Nearest Neighbor Algorithm. 2002	0.523
L. J. Shustek, An Algorithm for Finding Nearest Neighbors. 1975	0.523
F. Baskett, An Algorithm for Finding Nearest Neighbors. 1975	0.523
J. H. Friedman, An Algorithm for Finding Nearest Neighbors. 1975	0.523
H. Alt, The Nearest Neighbor. 2001	0.615
H. Samet, An efficient nearest neighbor algorithm for P2P settings. 2005	0.684
D. Nayar, An efficient nearest neighbor algorithm for P2P settings. 2005	0.684
E. Tanin, An efficient nearest neighbor algorithm for P2P settings. 2005	0.684
J. Ratsaby, An Incremental Nearest Neighbor Algorithm with Queries. 1997	0.684
T. G. Dietterich, Locally Adaptive Nearest Neighbor Algorithms. 1993	0.684
D. Wettschereck, Locally Adaptive Nearest Neighbor Algorithms. 1993	0.684
P. Zezula, A distributed incremental nearest neighbor algorithm. 2007	0.684
F. Rabitti, A distributed incremental nearest neighbor algorithm. 2007	0.684
C. Gennaro, A distributed incremental nearest neighbor algorithm. 2007	0.684
F. Falchi, A distributed incremental nearest neighbor algorithm. 2007	0.684
W. Tiller, Algorithm for finding all k nearest neighbors. 2002	0.684
L. A. Piegl, Algorithm for finding all k nearest neighbors. 2002	0.684
J. E. Hopcroft, A Note on Rabin's Nearest-Neighbor Algorithm. 1979	0.684

O resultado contém m tuplas para cada um dos documentos retornados. A mesma consulta considerando $k = 5$ e $w_{div} = 0,25$ para a consulta aos k-vizinhos diversos mais próximos resultou nas tuplas apresentadas na Tabela 5.3. O resultado contém tuplas mais interessantes por ter explorado a diversidade entre os elementos mais próximos, tendo retornado as 5 primeiras publicações distintas mais próximas do elemento de consulta.

Tabela 5.3: Consulta aos k-vizinhos diversos mais próximos. Elemento de consulta: "Nearest Neighbor Algorithms"

Elemento	δ
X. Zhang, Kernel Nearest Neighbor Algorithm. 2002	0.523
L. J. Shustek, An Algorithm for Finding Nearest Neighbors. 1975	0.523
H. Alt, The Nearest Neighbor. 2001	0.615
H. Samet, An efficient nearest neighbor algorithm for P2P settings. 2005	0.684
J. Ratsaby, An Incremental Nearest Neighbor Algorithm with Queries. 1997	0.684

5.3.2 Solução Exaustiva

Neste experimento foram comparados os algoritmos k-NDNq-Grasp-PR, k-NDNq-Grasp-EV and k-NDNq-Guloso em relação ao algoritmo k-NDNq-Exaustivo. O espaço de busca foi limitado em 200-vizinhos mais próximos, permitindo executar o algoritmo k-NDNq-Exaustivo considerando $k = 5$ vizinhos diversos mais próximos. A Tabela 5.4 apresenta o *gap* médio com relação ao algoritmo exaustivo considerando o peso da diversidade $w_{div} = 0,35$.

Esses resultados indicam que os algoritmos k-NDNq-Grasp-PR e k-NDNq-Grasp-EV alcançaram melhor qualidade que o algoritmo k-NDNq-Guloso e que estão próximos das soluções do algoritmo k-NDNq-Exaustivo. Ambos os algoritmos foram executados em uma fração do tempo requerido para executar a solução exaustiva. Para todos os conjuntos testados, as diferenças da estratégia gulosa para a exaustiva foram altas quando comparadas às diferenças do k-NDNq-Grasp-PR e k-NDNq-Grasp-EV em relação à estratégia exaustiva. Ambos k-NDNq-Grasp-PR e k-NDNq-Grasp-EV encontraram soluções muito próximas às soluções ótimas, com a medida *gap* variando entre $-0,3\%$ a $-5,4\%$. A Figura 5.2 apresenta os gráficos de perfis de desempenho para os experimentos realizados, que confirmam os resultados apresentados pela medida *gap*. Neles, as curvas dos algoritmos k-NDNq-Grasp-PR e k-NDNq-Grasp-EV estão bem próximas das curvas referentes às execuções do algoritmo exaustivo.

Tabela 5.4: Comparação do k-NDNq-Guloso e k-NDNq-Grasp com relação ao k-NDNq-Exaustivo.

Conjunto	Método	Gap médio %	Tempo médio (s)
Nasa	k-NDNq-Exaustivo	-	28,92
	k-NDNq-Grasp-PR	-5,36	0,01
	k-NDNq-Grasp-EV	-4,52	0,01
	k-NDNq-Guloso	-55,51	0,01
Cores	k-NDNq-Exaustivo	-	28,95
	k-NDNq-Grasp-PR	-1,06	0,02
	k-NDNq-Grasp-EV	-1,66	0,02
	k-NDNq-Guloso	-39,29	0,02
Faces	k-NDNq-Exaustivo	-	29,04
	k-NDNq-Grasp-PR	-0,30	0,10
	k-NDNq-Grasp-EV	-0,43	0,10
	k-NDNq-Guloso	-5,74	0,10
Documentos	k-NDNq-Exaustivo	-	29,29
	k-NDNq-Grasp-PR	-4,77	0,39
	k-NDNq-Grasp-EV	-4,14	0,39
	k-NDNq-Guloso	-48,56	0,42
DBLP	k-NDNq-Exaustivo	-	28,81
	k-NDNq-Grasp-PR	-3,30	0,01
	k-NDNq-Grasp-EV	-3,32	0,01
	k-NDNq-Guloso	-38,07	0,01

5.3.3 Número de elementos

Neste experimento, o objetivo é avaliar os resultados quando o número de elementos diversos k aumenta. O espaço de busca foi limitado em 1.000-vizinhos mais próximos, o que torna impraticável a execução do algoritmo exaustivo. A Tabela 5.5 apresenta o *gap* médio da estratégia k-NDNq-Grasp em relação ao algoritmo k-NDNq-Guloso considerando o peso da diversidade $w_{div} = 0,35$. Embora os resultados apresentados indiquem uma diminuição do ganho de qualidade da estratégia k-NDNq-Grasp em relação ao algoritmo k-NDNq-Guloso com o aumento do número de elementos diversos pedido, ainda assim a estratégia k-NDNq-Grasp se mantém como a que apresentou os menores valores para a função de pontuação. Por exemplo, nesse experimento o maior valor de *gap* entre essas estratégias foi de $125,27\%$ para o conjunto DBLP com $k = 10$ e o menor valor de *gap* foi de $5,46\%$ para o conjunto Faces com $k = 10$.

Além disso, é importante notar que o tempo necessário para a execução da estratégia k-NDNq-Guloso aumenta consideravelmente em relação ao algoritmo k-NDNq-Grasp com o aumento do número de elementos pedido. A Tabela 5.6 apresenta os tempos médios de execução das consultas apresentadas na Tabela 5.5.

A Figura 5.3 apresenta os gráficos de perfis de desempenho para os experimentos re-

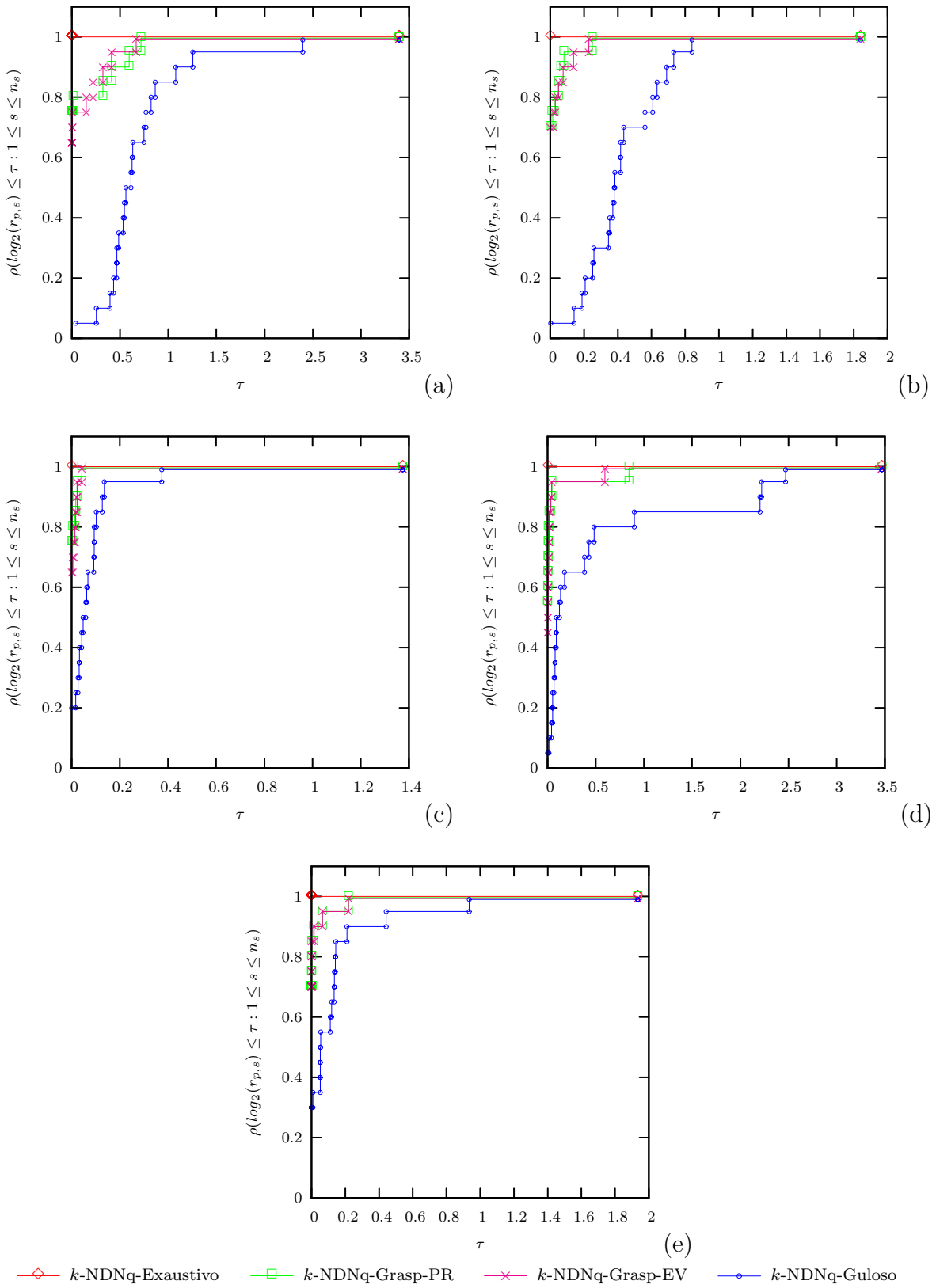

Figura 5.2: Gráficos de perfis de desempenho considerando as estratégias exaustiva, Grasp e Gulosa. (a) Conjunto Nasa. (b) Conjunto Cores. (c) Conjunto Faces. (d) Conjunto Documentos. (e) Conjunto DBLP.

Tabela 5.5: Avaliação do parâmetro k. Comparação do k-NDNq-Grasp com relação ao k-NDNq-Guloso. *Gap* médio em %.

Conjunto	Método	k (*gap* médio %)				
		10	20	30	40	50
Nasa	k-NDNq-Grasp-PR	30,93	26,68	24,69	24,68	24,60
	k-NDNq-Grasp-EV	31,63	26,81	24,87	24,78	23,72
Cores	k-NDNq-Grasp-PR	17,54	16,34	17,15	16,35	16,24
	k-NDNq-Grasp-EV	17,44	16,34	17,25	16,39	16,27
Faces	k-NDNq-Grasp-PR	5,46	6,46	6,76	6,71	6,54
	k-NDNq-Grasp-EV	5,47	6,51	6,77	6,75	6,57
Documentos	k-NDNq-Grasp-PR	30,18	22,39	20,82	20,42	19,85
	k-NDNq-Grasp-EV	30,26	22,46	20,85	20,52	19,89
DBLP	k-NDNq-Grasp-PR	124,99	51,36	40,38	37,92	33,92
	k-NDNq-Grasp-EV	125,27	51,81	40,72	38,24	34,44

Tabela 5.6: Avaliação do parâmetro k. Comparação do k-NDNq-Grasp com relação ao k-NDNq-Guloso. Tempo de execução médio em segundos.

Conjunto	Método	k (tempo médio em segundos)				
		10	20	30	40	50
Nasa	k-NDNq-Guloso	0,27	2,23	7,56	17,85	34,63
	k-NDNq-Grasp-PR	0,45	0,61	0,81	1,10	1,39
	k-NDNq-Grasp-EV	0,89	1,56	2,29	3,10	3,99
Cores	k-NDNq-Guloso	1,43	11,97	40,72	96,32	187,00
	k-NDNq-Grasp-PR	1,76	1,92	2,12	2,38	2,69
	k-NDNq-Grasp-EV	2,21	2,86	3,59	4,38	5,30
Faces	k-NDNq-Guloso	9,42	78,76	268,00	634,00	1.235,00
	k-NDNq-Grasp-PR	10,76	10,92	11,14	11,39	11,70
	k-NDNq-Grasp-EV	11,11	11,66	12,29	13,01	13,82
Documentos	k-NDNq-Guloso	18,62	155,00	529,00	1.240,00	2.395,00
	k-NDNq-Grasp-PR	18,35	18,49	18,66	18,92	19,27
	k-NDNq-Grasp-EV	18,76	19,36	20,13	20,89	21,80
DBLP	k-NDNq-Guloso	0,70	5,53	18,32	43,21	84,33
	k-NDNq-Grasp-PR	0,90	1,02	1,17	1,38	1,61
	k-NDNq-Grasp-EV	1,11	1,45	1,85	2,30	2,86

alizados referentes à configuração com $k = 30$, que confirmam os resultados apresentados pela medida *gap*. Neles, as curvas dos algoritmos k-NDNq-Grasp-PR e k-NDNq-Grasp-EV apresentaram melhor desempenho que o k-NDNq-Guloso, que em geral obteve probabilidade igual a 1 com fator τ em torno de 50% da melhor razão de desempenho possível para os conjuntos Documentos e DBLP, em torno de 30% para os conjuntos Nasa e Cores e de 13% para o conjunto Faces.

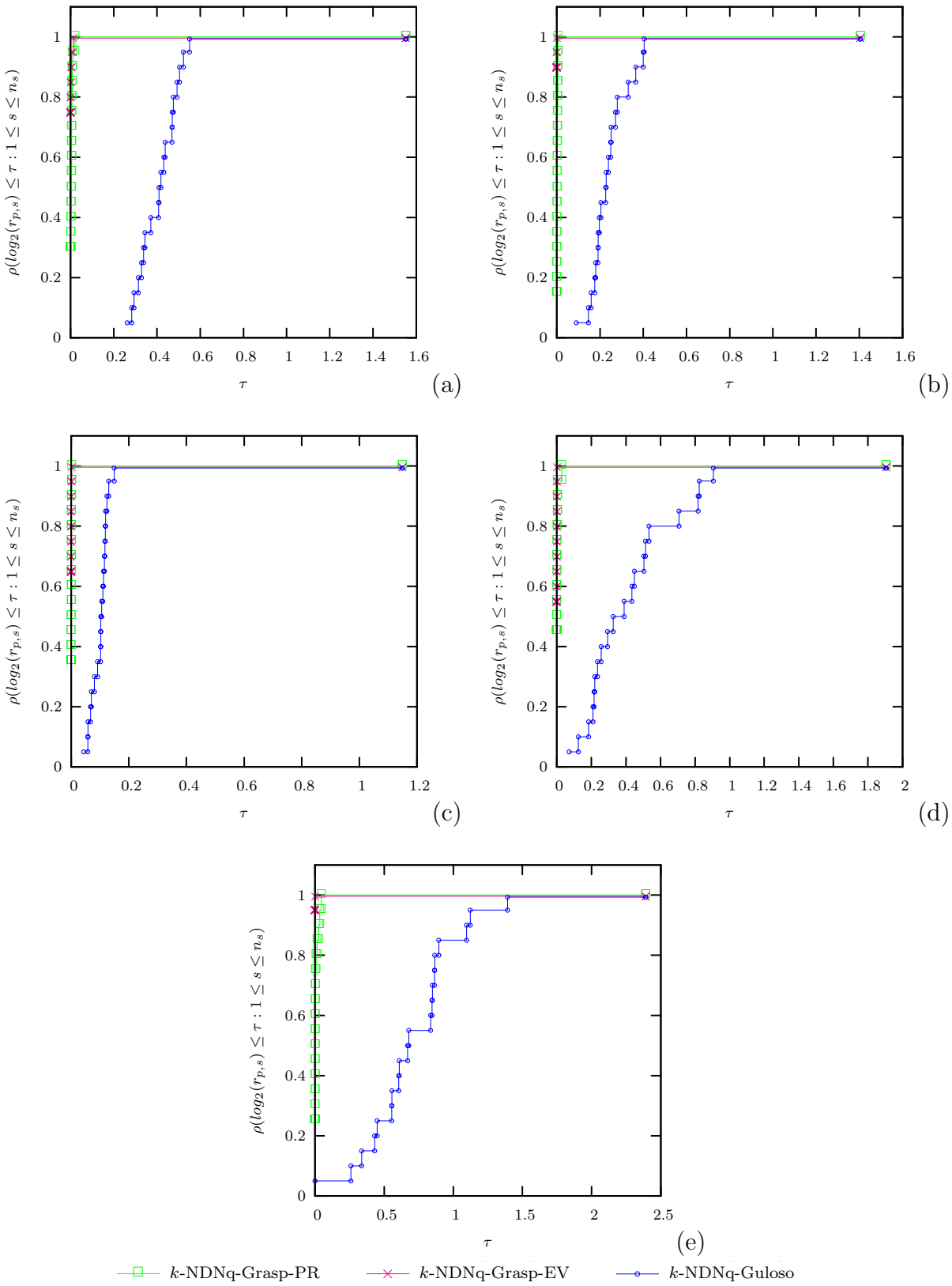

Figura 5.3: Gráficos de perfis de desempenho considerando as estratégias Grasp e Gulosa referentes à configuração com $k = 30$. (a) Conjunto Nasa. (b) Conjunto Cores. (c) Conjunto Faces. (d) Conjunto Documentos. (e) Conjunto DBLP.

5.3.4 Aumento do Espaço de Busca

Neste experimento, o objetivo é avaliar o comportamento dos métodos propostos com o aumento do espaço de busca. A Tabela 5.7 apresenta o *gap* médio da estratégia k-NDNq-Grasp em relação ao algoritmo k-NDNq-Guloso considerando o peso da diversidade $w_{div} = 0,3$ e $k = 20$ elementos diversos. A Tabela 5.8 apresenta o tempo médio em segundos para a execução das consultas apresentadas na Tabela 5.7.

Tabela 5.7: Aumento do espaço de busca. Comparação do k-NDNq-Grasp com relação ao k-NDNq-Guloso. *Gap* médio em %.

Conjunto	Método	limite ℓ (*gap* médio %)				
		400	800	1.200	1.600	2.000
Nasa	k-NDNq-Grasp-PR	22,58	23,89	24,44	24,55	24,75
	k-NDNq-Grasp-EV	22,80	24,23	24,62	24,87	24,99
Cores	k-NDNq-Grasp-PR	16,82	17,03	17,12	17,13	17,25
	k-NDNq-Grasp-EV	16,88	17,12	17,19	13,32	17,36
Faces	k-NDNq-Grasp-PR	5,93	5,94	5,91	-	-
	k-NDNq-Grasp-EV	5,99	5,96	5,96	-	-
Documentos	k-NDNq-Grasp-PR	21,53	22,50	22,91	23,18	23,52
	k-NDNq-Grasp-EV	21,54	22,55	23,00	23,24	23,69
DBLP	k-NDNq-Grasp-PR	55,74	57,05	59,49	59,67	59,76
	k-NDNq-Grasp-EV	55,83	57,29	59,50	60,07	60,24

Tabela 5.8: Aumento do espaço de busca. Comparação do k-NDNq-Grasp com relação ao k-NDNq-Guloso. Tempo de execução médio em segundos.

Conjunto	Método	limite ℓ (tempo médio em segundos)				
		400	800	1.200	1.600	2.000
Nasa	k-NDNq-Guloso	0,86	1,76	2,67	3,59	4,47
	k-NDNq-Grasp-PR	0,11	0,40	0,84	1,44	2,21
	k-NDNq-Grasp-EV	0,27	0,99	2,15	3,73	5,71
Cores	k-NDNq-Guloso	4,66	9,53	14,43	19,32	24,23
	k-NDNq-Grasp-PR	0,33	1,25	2,74	4,82	7,49
	k-NDNq-Grasp-EV	0,48	1,84	4,03	7,06	10,93
Faces	k-NDNq-Guloso	30,65	62,74	80,00	-	-
	k-NDNq-Grasp-PR	1,76	7,00	11,26	-	-
	k-NDNq-Grasp-EV	1,91	7,52	12,01	-	-
Documentos	k-NDNq-Guloso	71,81	147,00	222,00	297,00	373,00
	k-NDNq-Grasp-PR	3,74	14,59	32,45	57,27	88,63
	k-NDNq-Grasp-EV	3,90	15,20	33,84	59,66	92,27
DBLP	k-NDNq-Guloso	2,04	4,22	6,36	8,57	10,70
	k-NDNq-Grasp-PR	0,17	0,65	1,43	2,54	3,96
	k-NDNq-Grasp-EV	0,25	0,93	2,05	3,62	5,63

A análise dos resultados apresentados na Tabela 5.7 permite perceber que a estratégia k-NDNq-Grasp resultou em qualidade constante com o aumento do espaço de busca para

todos os conjuntos de dados avaliados. Além disso, a análise dos resultados descritos na Tabela 5.8 permite verificar que a estratégia k-NDNq-Grasp resultou em tempos de execução em geral menores que do algoritmo k-NDNq-Guloso. Por exemplo, nesse experimento o maior valor de *gap* entre essas estratégias foi de $60,24\%$ para o conjunto DBLP com limite $\ell = 2.000$ e o menor valor de *gap* foi de $5,93\%$ para o conjunto Faces com $\ell = 400$.

5.4 Considerações Finais

Este capítulo apresentou as consultas aos k-vizinhos diversos mais próximos, que têm por objetivo prover diversidade nos resultados de consultas aos k-vizinhos mais próximos, visando a melhoria da semântica nas consultas por conteúdo de imagens. O modelo considerou os dados como imersos em espaços métricos, de modo que funções de distância são empregadas para a computação da dissimilaridade e da diversidade, uma vez que as principais consultas realizadas sobre grandes conjuntos de imagens levam em consideração a dissimilaridade entre suas características. O problema proposto tem complexidade de tempo polinomial com expoente k, logo foram propostas heurísticas baseadas em busca aleatória para encontrar soluções aproximadas para o problema em tempo real, bem como um algoritmo guloso que resulta em soluções determinísticas. Os algoritmos propostos mostraram-se eficientes em experimentos realizados com conjuntos de dados reais com alta dimensionalidade. A avaliação do tratamento da diversidade em consultas por similaridade na recuperação de imagens por conteúdo é tema dos trabalhos futuros descritos no próximo capítulo.

Capítulo

6

Conclusão

"If I have seen farther than others,
it is because I have stood on the shoulders of giants"
- Isaac Newton.

A recuperação de imagens por conteúdo é uma tarefa almejada por várias aplicações, entre elas os sistemas de ensino e auxílio ao diagnóstico médico. Nessas técnicas, são empregados vetores de características extraídos automaticamente das imagens. Entretanto a recuperação baseada em vetores de características pode resultar em imagens que não atendem ao desejo do usuário. A redução da descontinuidade semântica entre a percepção humana e as imagens resultantes de consultas por similaridade é uma área de pesquisa que ainda necessita de pesquisa e desenvolvimento. Entre as técnicas empregadas para melhorar o estado da arte, os métodos de realimentação de relevância têm por objetivo permitir que o sistema adapte as consultas subseqüentes à intenção do usuário, com processamento em tempo real.

Em pouco mais de uma década, diversas técnicas de realimentação de relevância foram propostas para tratar da descontinuidade semântica em buscas por conteúdo de imagens. Todavia, nenhum trabalho havia abordado a modelagem e a otimização das consultas de múltiplos centros envolvendo tanto o número de elementos desejados quanto o raio de abrangência em espaços métricos com respostas exatas. O tratamento dessas consultas em espaços métricos é importante para permitir a manipulação eficiente de vetores de alta dimensionalidade ou de conjuntos de dados que não apresentam o conceito de dimensão. Essas consultas são de grande utilidade e têm aplicação direta em métodos de realimentação de relevância baseados na movimentação de múltiplos centros de consulta.

Outro ponto importante que não havia sido abordado na literatura considerando espaços métricos está relacionado com a diversidade entre os elementos resultantes de consultas aos vizinhos mais próximos baseado apenas em cálculos de distância, além da semântica associada a essa diversidade. O tratamento da diversidade nesses espaços é importante uma vez que tem potencial para melhorar a qualidade dos resultados.

As contribuições desta tese têm como objetivo melhorar o estado da arte dessas técnicas, especialmente com relação à escalabilidade dos métodos e a qualidade dos resultados. Os pontos centrais foram a definição e a otimização das consultas de múltiplos centros, denominadas consultas por similaridade agregada, e o tratamento da diversidade em consultas por similaridade. A seguir são sintetizadas as principais contribuições.

6.1 Principais Contribuições

Este trabalho apresenta as seguintes contribuições:

- Definição das consultas por similaridade agregada como uma generalização das consultas por similaridade baseadas em um único centro e a aplicação dessas consultas em métodos de realimentação de relevância em consultas por conteúdo de imagens, incluindo:
  - as consultas limitadas por k elementos e as consultas limitadas por raio ξ;
  - a propriedade do raio agregado mínimo;
  - o uso de pesos para indicar o grau de relevância nessas técnicas;
  - a avaliação empírica em comparação com a técnica de Rocchio.
- Desenvolvimento de um método exato denominado *Metric Aggregate Similarity Search (MASS)*, para otimização de consultas por similaridade agregada em métodos de acesso métricos, incluindo:
  - a generalização para os fatores de agregação g no intervalo $-\infty \leq g \leq \infty \wedge g \neq 0$;
  - os algoritmos para execução eficiente das consultas limitadas por k e por raio;
  - a avaliação empírica com relação aos métodos Falcon e R-tree MBM.
- Modelo para tratamento da diversidade em consultas por similaridade em espaços métricos, incluindo:
  - estudo da complexidade e apresentação do algoritmo para solução exata;

- heurísticas para sua otimização baseadas em métodos de buscas aleatórias e em uma abordagem gulosa;
- a avaliação empírica das heurísticas com relação à qualidade dos resultados encontrados e ao tempo de execução das consultas.

Um ponto importante das técnicas propostas é que elas foram implementadas nos métodos de acesso existentes, não exigindo a inclusão de novas informações nos métodos de acesso. As técnicas foram integradas aos métodos de acesso métrico M-tree e Slim-tree e podem ser integradas a outros métodos de acesso métrico, como a DBM-tree [Vieira et al., 2006].

6.2 Publicações

Os seguintes trabalhos contêm contribuições parciais desta tese:

- Razente, H. L., Barioni, M. C. N., Traina, A. J. M., Faloutsos, C., e Traina-Jr., C. (2008). A novel optimization approach to efficiently process aggregate similarity queries in metric access methods. In *ACM International Conference on Information and Knowledge Management (CIKM)*, páginas 193-202, Napa Valley, California. DOI: 10.1145/1458082.1458110;
- Razente, H. L., Barioni, M. C. N., Traina, A. J. M., e Traina Jr, C. (2008). Aggregate similarity queries in relevance feedback methods for content-based image retrieval. In *ACM Symposium on Applied Computing (SAC)*, páginas 869-874, Fortaleza (CE). DOI: 10.1145/1363686.1363887;
- Barioni, M. C. N., Razente, H. L., Traina, A. J. M., Traina Jr, C. (2008). Seamlessly integrating similarity queries in SQL. *Software, Practice & Experience*, 39(4):355-384, DOI: 10.1002/spe.898;
- Razente, H. L., Barioni, M. C. N., Traina, A. J. M., Traina Jr, C. (2007). Constrained Aggregate Similarity Queries in Metric Spaces. In *Simpósio Brasileiro de Banco de Dados (SBBD)*, páginas 145-159, João Pessoa (PB). SBC;
- Razente, H. L., Barioni, M. C. N., Traina, A. J. M., Traina Jr, C. (2007). Consultas por similaridade agregada em métodos de realimentação de relevância para consultas por conteúdo de imagens. In *I Sessão de Pôsteres do Simpósio Brasileiro de Banco de Dados (SBBD)*, páginas 3-6, João Pessoa (PB). SBC;
- Razente, H. L., Barioni, M. C. N., Traina, A. J. M., e Traina Jr, C. (2006). Recuperação de imagens médicas por conteúdo em um sistema de gerenciamento de

banco de dados de código livre. In *X Congresso Brasileiro de Informática em Saúde (CBIS)*, páginas 1561-1566, Florianópolis (SC). Sociedade Brasileira de Informática em Saúde (SBIS);

- Barioni, M. C. N., Razente, H. L., Traina, A. J. M., Traina Jr, C. (2006). SIREN: A Similarity Retrieval Engine For Complex Data. In *Demonstration Session of the International Conference on Very Large Data Bases (VLDB)*, páginas 1155-1158, Seul, Coréia do Sul.

6.3 Propostas para Trabalhos Futuros

As contribuições apresentadas nesta tese geraram a necessidade de novos estudos, tanto para estender as técnicas desenvolvidas quanto para abordar outros fatores. A integração das técnicas de realimentação de relevância com os sistemas de gerenciamento de banco de dados é uma importante questão em aberto que permitirá o desenvolvimento de aplicações com maior qualidade semântica, nas quais essas técnicas possam ser empregadas para a diminuição da descontinuidade semântica na recuperação de imagens.

A seguir são apresentadas as propostas de trabalhos futuros:

- Avaliação da semântica e da possibilidade de otimização de outras funções de agregação na computação da similaridade agregada, como média harmônica e mediana;
- Treinamento de funções de distância para domínios específicos de imagens com base no método de realimentação de relevância que utiliza as consultas por similaridade agregada;
- Estudo e desenvolvimento de buscas tabu (*tabu search*) e algoritmos genéticos para otimização das consultas aos k-vizinhos diversos mais próximos;
- Integração dos métodos propostos com consultas *top-k* para consultas complexas envolvendo documentos de textos e similaridade de imagens para, por exemplo, a integração de recuperação por conteúdo de imagens de exames com laudos textuais realizados por médicos especialistas;
- avaliação da semântica das consultas aos vizinhos mais próximos diversos na recuperação de imagens e estudo da diversidade nas consultas por similaridade agregada;
- Extensão do mecanismo de execução de consultas por conteúdo de imagens denominado SIREN (*Similarity Retrieval Engine*) [Barioni et al., 2006], com uma proposta de linguagem de consulta que permita a realimentação de relevância dos usuários;

- Estudo da integração das técnicas de realimentação de relevância com perfis de usuários;
- Tratamento da diversidade em consultas com múltiplos centros;
- Avaliação do uso de funções de distância distintas para similaridade e diversidade;
- Tratamento de buscas com múltiplas funções de distância ao invés de múltiplos centros, ou com múltiplos centros e múltiplas funções de distância, e correspondente avaliação como mecanismo de realimentação de relevância.

Referências Bibliográficas

[Agrawal et al., 2009] Agrawal, R., Gollapudi, S., Halverson, A., e Ieong, S. (2009). Diversifying search results. In *ACM International Conference on Web Search and Data Mining (WSDM)*, páginas 5–14, Barcelona, Spain. ACM, DOI: 10.1145/1498759.1498766.

[Ahn et al., 2001] Ahn, H. K., Mamoulis, N., Wong, H. M., e Wong, H. M. (2001). A survey on multidimensional access methods. Technical report, UU-CS-2001-14, University of Science and Technology, Clearwater Bay, Hong Kong, 19 páginas, http://www.scientificcommons.org/42685195.

[Altavista, 2009] Altavista (2009). Altavista image search. http://www.altavista.com.

[Alto et al., 2005] Alto, H., Rangayyan, R. M., e Desautels, J. E. L. (2005). Content-based retrieval and analysis of mammographic masses. *Journal of Electronic Imaging*, 14(2):1–17.

[Androutsos et al., 1998] Androutsos, D., Plataniotis, K. N., e Venetsanopoulos, A. N. (1998). Distance measures for color image retrieval. In *International Conference on Image Processing (ICIP)*, volume 2, páginas 770–774, Chicago, IL. IEEE, DOI: 10.1109/ICIP.1998.723652.

[Antani et al., 2006] Antani, S., Cheng, J., Long, J., Long, L. R., e Thoma, G. R. (2006). Medical validation and cbir of spine x-ray images over the internet. In *IS&T/SPIE Electronic Imaging Science and Technology 2006: Internet Imaging VII. SPIE Vol. 6061*, páginas 1–9, San Jose, CA. SPIE, DOI: 10.1117/12.649372.

[Arivazhagan e Ganesan, 2003] Arivazhagan, S. e Ganesan, L. (2003). Texture classification using wavelet transform. *Pattern Recognition Letters*, 24:1513–1521, DOI: 10.1016/S0167-8655(02)00390-2.

[Asuncion e Newman, 2007] Asuncion, A. e Newman, D. J. (2007). UCI machine learning repository. University of California, Irvine, http://archive.ics.uci.edu/ml/.

[Baeza-Yates e Ribeiro-Neto, 1999] Baeza-Yates, R. e Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley, Wokingham, UK.

[Balan et al., 2005] Balan, A. G. R., Traina, A. J. M., Traina-Jr, C., e Marques, P. M. A. (2005). Fractal analysis of image textures for indexing and retrieval by content. In *IEEE International Symposium on Computer-Based Medical Systems (CBMS)*, páginas 22–24, Dublin, Irlanda. IEEE, DOI: 10.1109/CBMS.2005.54.

[Barioni et al., 2006] Barioni, M. C. N., Razente, H. L., Traina, A. J. M., e Traina-Jr, C. (2006). Siren: A similarity retrieval engine for complex data. In *Demonstration Session of the International Conference on Very Large Data Bases (VLDB)*, páginas 1155–1158, Seul, Coréia do Sul. ACM.

[Barnard e Bayes, 1958] Barnard, G. A. e Bayes, T. (1958). Studies in the history of probability and statistics: IX. Thomas Bayes's essay towards solving a problem in the doctrine of chances. *Biometrika*, 45(3/4):293–315.

[Bayes, 1763] Bayes, T. (1763). An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, F. R. S. communicated by Mr. Price, in a letter to John Canton, A. M. F. R. S. *Philosophical Transactions, Giving Some Account of the Present Undertakings, Studies and Labours of the Ingenious in Many Considerable Parts of the World*, 53:370–418.

[Bazaraa et al., 2004] Bazaraa, M. S., Jarvis, J. H., e Sherali, H. D. (2004). *Linear Programming and Network Flows*. Wiley.

[Beckmann et al., 1990] Beckmann, N., Kriegel, H.-P., Schneider, R., e Seeger, B. (1990). The r*-tree: An efficient and robust access method for points and rectangles. In *International Conference on Management of Data (SIGMOD)*, páginas 322–331, Atlantic City, NJ. ACM, DOI: 10.1145/93597.98741.

[Bin e Jia-Xiong, 2002] Bin, Y. e Jia-Xiong, P. (2002). Invariance analysis of improved zernike moments. *Journal of Optics A: Pure and Applied Optics*, 4:606–614, DOI: 10.1088/1464-4258/4/6/304.

[Binderberger e Mehrotra, 2003] Binderberger, M. O. e Mehrotra, S. (2003). *Relevance Feedback in Multimedia Databases. In Handbook of Video Databases: Design and Applications*, capítulo 23, páginas 1–28. CRC Press.

[Böhm et al., 2001] Böhm, C., Berchtold, S., e Keim, D. A. (2001). Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases. *ACM Computing Surveys (CSUR)*, 33(3):322–373, ACM, DOI: 10.1145/502807.502809.

[Boser et al., 1992] Boser, B. E., Guyon, I. M., e Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *ACM Workshop on Computational Learning Theory*, páginas 144–152, Pittsburgh, PA. ACM.

[Bozkaya e Özsoyoglu, 1997] Bozkaya, T. e Özsoyoglu, M. (1997). Distance-based indexing for high-dimensional metric spaces. In *International Conference on Management of Data (SIGMOD)*, páginas 357–368, Tucson, AZ.

[Bugatti et al., 2008] Bugatti, P. H., Traina, A. J. M., e Traina-Jr, C. (2008). Assessing the best integration between distance-function and image-feature to answer similarity queries. In *ACM Symposium on Applied Computing (SAC)*, páginas 1225–1230, Fortaleza, CE. ACM, DOI: 10.1145/1363686.1363969.

[Carson et al., 2002] Carson, C., Belongie, S., Greenspan, H., e Malik, J. (2002). Blobworld: image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 24(8):1026–1038, DOI: 10.1109/TPAMI.2002.1023800.

[Chang e Fu, 1980] Chang, N.-S. e Fu, K.-S. (1980). Query-by pictorial-example. *IEEE Transactions on Software Engineering*, 6(6):519–524, DOI: 10.1109/TSE.1980.230801.

[Chang e Kunii, 1981] Chang, S.-K. e Kunii, T. L. (1981). Pictorial data-base systems. *IEEE Computer*, 14(11):13–21, DOI: 10.1109/C-M.1981.220243.

[Chang e Liu, 1984] Chang, S.-K. e Liu, S.-H. (1984). Picture indexing and abstraction techniques for pictorial databases. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 6:475–484, DOI: 10.1109/TPAMI.1984.4767552.

[Chen et al., 2003] Chen, Y., Wang, J. Z., e Krovetz, R. (2003). An unsupervised learning approach to content-based image retrieval. In *International Symposium on Signal Processing and Its Applications (ISSPA)*, páginas 197–200, Paris, França. IEEE, DOI: 10.1109/ISSPA.2003.1224674.

[Chen et al., 2001] Chen, Y., Zhou, X., e Huang, T. S. (2001). One-class SVM for learning in image retrieval. In *International Conference on Image Processing (ICIP)*, páginas 34–37, Thessaloniki, Grécia. IEEE, DOI: 10.1109/ICIP.2001.958946.

[Ciaccia et al., 1997] Ciaccia, P., Patella, M., e Zezula, P. (1997). M-tree: An efficient access method for similarity search in metric spaces. In *International Conference on Very Large Data Bases (VLDB)*, páginas 426–435, Atenas, Grécia. Morgan Kaufmann.

[Clark e Niblett, 1989] Clark, P. e Niblett, T. (1989). The CN2 induction algorithm. *Machine Learning*, 3(4):261–283, Kluwer, DOI: 10.1023/A:1022641700528.

[Comer e Delp, 2000] Comer, M. L. e Delp, E. J. (2000). The EM/MPM algorithm for segmentation of textured images: Analysis and further experimental results. *IEEE Transactions on Image Processing*, 9(10):1731–1744, DOI: 10.1109/83.869185.

[Datta et al., 2008] Datta, R., Joshi, D., Li, J., e Wang, J. Z. (2008). Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (CSUR)*, 40(2):1–60, ACM, DOI: 10.1145/1348246.1348248.

[Daubechies, 1990] Daubechies, I. (1990). The wavelet transform, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory*, 36(5):961–1005, DOI: 10.1109/18.57199.

[Davis e Goadrich, 2006] Davis, J. e Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *International Conference on Machine Learning (ICML)*, páginas 233–240, Pittsburgh, PA. ACM, DOI: 10.1145/1143844.1143874.

[Deng e Manjunath, 2001] Deng, Y. e Manjunath, B. S. (2001). Unsupervised segmentation of color-texture regions in images and video. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 23(8):800–810, DOI: 10.1109/34.946985.

[Dolan e Moré, 2002] Dolan, E. D. e Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91(2):201–213, Springer, DOI: 10.1007/s101070100263.

[Doulamis e Doulamis, 2004] Doulamis, A. e Doulamis, N. (2004). Generalized nonlinear relevance feedback for interactive content-based retrieval and organization. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(5):656–671, DOI: 10.1109/TCSVT.2004.826752.

[Doulamis e Doulamis, 2006] Doulamis, N. e Doulamis, A. (2006). Evaluation of relevance feedback schemes in content-based in retrieval systems. *Signal Processing: Image Communication*, 21(4):334–357, DOI: 10.1016/j.image.2005.11.006.

[Dua et al., 2009] Dua, S., Singh, H., e Thompson, H. W. (2009). Associative classification of mammograms using weighted rules. *Expert Systems with Applications*, 36(5):9250–9259, Pergamon Press, Inc., DOI: 10.1016/j.eswa.2008.12.050.

[Dy et al., 2003] Dy, J. G., Brodley, C. E., Kak, A., Broderick, L. S., e Aisen, A. M. (2003). Unsupervised feature selection applied to content-based retrieval of lung images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 25(3):373–378, DOI: 10.1109/TPAMI.2003.1182100.

[Eakins e Graham, 1999] Eakins, J. e Graham, M. (1999). Content-based image retrieval. Technical report, JISC Technology Application Program, University of Northumbria at Newcastle, n^o 39.

[Euzenat e Shvaiko, 2007] Euzenat, J. e Shvaiko, P. (2007). *Ontology Matching*. Springer-Verlag.

[Faloutsos e Lin, 1995] Faloutsos, C. e Lin, K.-I. D. (1995). Fastmap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In *International Conference on Management of Data (SIGMOD)*, páginas 163–174, Zurich, Switzerland.

[Fayyad e Uthurusamy, 2002] Fayyad, U. e Uthurusamy, R. (2002). Evolving data mining into solutions for insights. *Communications of the ACM*, 45(8):28–31, DOI: 10.1145/545151.545174.

[Felipe et al., 2006] Felipe, J. C., Ribeiro, M. X., Sousa, E. P. M., Traina, A. J. M., e Traina-Jr, C. (2006). Effective shape-based retrieval and classification of mammograms. In *ACM Symposium on Applied Computing (SAC)*, páginas 250–255, Dijon, França. ACM, DOI: 10.1145/1141277.1141333.

[Felipe et al., 2009] Felipe, J. C., Traina-Jr, C., e Traina, A. J. M. (2009). A new family of distance functions for perceptual similarity retrieval of medical images. *Journal of Digital Imaging*, 22(2):183–201, DOI: 10.1007/s10278-007-9084-x.

[Feng et al., 2001] Feng, H., Castanon, D. A., e Karl, W. C. (2001). A curve evolution approach for image segmentation using adaptiveflows. In *IEEE International Conference on Computer Vision (ICCV)*, páginas 494–499, Vancouver, Canadá. IEEE, DOI: 10.1109/ICCV.2001.937666.

[Feo e Resende, 1995] Feo, T. A. e Resende, M. G. C. (1995). Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6:109–133, DOI: 10.1007/BF01096763.

[Ferecatu et al., 2005] Ferecatu, M., Crucianu, M., e Boujemaa, N. (2005). Improving performance of interactive categorization of images using relevance feedback. In *International Conference on Image Processing (ICIP)*, volume 1, páginas 1197–1200, Genova, Itália. IEEE, DOI: 10.1109/ICIP.2005.1529971.

[Ferrari et al., 2004] Ferrari, R. J., Rangayyan, R. M., Desautels, J. E. L., Borges, R. A., e Frère, A. F. (2004). Automatic identification of the pectoral muscle in mammograms. *IEEE Transactions on Medical Imaging*, 23(2):232–245, DOI: 10.1109/TMI.2003.823062.

[Figueroa et al., 2009] Figueroa, K., Navarro, G., e Chavez, E. (2009). Metric spaces library. Disponível em http://sisap.org/Metric_Space_Library.html.

[Fleurent e Glover, 1999] Fleurent, C. e Glover, F. (1999). Improved constructive multistart strategies for the quadratic assignment problem using adaptive memory. *Informs Journal on Computing*, 11(2):198–204, Informs, DOI: 10.1287/ijoc.11.2.198.

[French et al., 2004] French, J. C., Jin, X., e Martin, W. N. (2004). An empirical investigation of the scalability of a multiple viewpoint cbir system. In *International Conference on Image and Video Retrieval (CIVR)*, volume 3115 do *Lecture Notes in Computer Science*, páginas 252–260, Dublin, Irlanda. Springer, DOI: 10.1007/b98923.

[Gabor, 1946] Gabor, D. (1946). Theory of communication. *Journal of the Institute of Electrical Engineers (London)*, 93-III(26):429–457.

[Gaede e Günther, 1998] Gaede, V. e Günther, O. (1998). Multidimensional access methods. *ACM Computing Surveys (CSUR)*, 30(2):170–231, ACM, DOI: 10.1145/280277.280279.

[Galassi et al., 2006] Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Booth, M., e Rossi, F. (2006). *GNU Scientific Library, Reference Manual, Edition 1.8 for GSL Version 1.8*. Network Theory Ltd.

[Geusebroek et al., 2005] Geusebroek, J.-M., Burghouts, G. J., e Smeulders, A. W. M. (2005). The Amsterdam library of object images. *International Journal of Computer Vision*, 61(1):103–112, DOI: 10.1023/B:VISI.0000042993.50813.60.

[Ghosh, 1996] Ghosh, J. B. (1996). Computational aspects of the maximum diversity problem. *Operations Research Letters*, 19(4):175–181, DOI: 10.1016/0167-6377(96)00025-9.

[Grundland e Dodgson, 2007] Grundland, M. e Dodgson, N. A. (2007). Decolorize: Fast, contrast enhancing, color to grayscale conversion. *Pattern Recognition*, 40(11):2891–2896, Elsevier, DOI: 10.1016/j.patcog.2006.11.003.

[Gu et al., 2002] Gu, J., Shu, H. Z., Toumoulin, C., e Luo, L. M. (2002). A novel algorithm for fast computation of zernike moments. *Pattern Recognition*, 35(12):2905–2911, DOI: 10.1016/S0031-3203(01)00194-7.

[Guha et al., 1998] Guha, S., Rastogi, R., e Shim, K. (1998). Cure: an efficient clustering algorithm for large databases. In *International Conference on Management of Data (SIGMOD)*, páginas 73–84, New York, NY. ACM, DOI: 10.1145/276304.276312.

[Guild, 1931] Guild, J. (1931). The colorimetric properties of the spectrum. *Philosophical Transactions of the Royal Society of London*, A230:149–187.

[Guttman, 1984] Guttman, A. (1984). R-trees: A dynamic index structure for spatial searching. In *International Conference on Management of Data (SIGMOD)*, páginas 47–57, Boston, MA.

[Hair-Jr et al., 1995] Hair-Jr, J. F., Anderson, R. E., Tatham, R. L., e Black, W. C. (1995). *Multivariate Data Analysis*. Prentice Hall, New Jersey, 5^a .

[Han e Kamber, 2006] Han, J. e Kamber, M. (2006). *Data mining: concepts and techniques*. Morgan Kaufmann, San Francisco, CA, 2^a edição .

[Haralick, 1979] Haralick, R. M. (1979). Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804.

[Haralick et al., 1973] Haralick, R. M., Shanmugarn, K., e Dinstein, I. (1973). Texture features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6):610–621.

[Heesch, 2008] Heesch, D. (2008). A survey of browsing models for content based image retrieval. *Multimedia Tools and Applications*, 40(2):261–284, Kluwer, DOI: 10.1007/s11042-008-0207-2.

[Hjaltason e Samet, 1995] Hjaltason, G. R. e Samet, H. (1995). Ranking in spatial databases. In *International Symposium on Advances in Spatial Databases (SSD)*, páginas 83–95, Portland, Maine.

[Hjaltason e Samet, 2003] Hjaltason, G. R. e Samet, H. (2003). Index-driven similarity search in metric spaces. *ACM Transactions on Database Systems (TODS)*, 28(4):517–580, ACM, DOI: 10.1145/958942.958948.

[Hoi et al., 2006] Hoi, S. C. H., Lyu, M. R., e Jin, R. (2006). A unified log-based relevance feedback scheme for image retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 18(4):509–524, DOI: 10.1109/TKDE.2006.53.

[Hua et al., 2006] Hua, K. A., Yu, N., e Liu, D. (2006). Query decomposition: A multiple neighborhood approach to relevance feedback processing in content-based image retrieval. In *International Conference on Data Engineering (ICDE)*, páginas 84–94, Atlanta, GA. DOI: 10.1109/ICDE.2006.123.

[Huang et al., 1997] Huang, J., Kumar, S. R., Mitra, M., Zhu, W.-J., e Zabih, R. (1997). Image indexing using color correlograms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 762–768, San Juan, Porto Rico. IEEE.

[Ilyas et al., 2008] Ilyas, I. F., Beskales, G., e Soliman, M. A. (2008). A survey of top-k query processing techniques in relational database systems. *ACM Computing Surveys (CSUR)*, 40(4):1–58, ACM, DOI: 10.1145/1391729.1391730.

[Ishikawa et al., 1998] Ishikawa, Y., Subramanya, R., e Faloutsos, C. (1998). Mindreader: Query databases through multiple examples. In *International Conference on Very Large Databases (VLDB)*, páginas 218–227, New York, NY.

[Jacobs et al., 2000] Jacobs, D. W., Weinshall, D., e Gdalyahu, Y. (2000). Classification with nonmetric distances: Image retrieval and class representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22(6):583–600, DOI: 10.1109/34.862197.

[Jain et al., 2004] Jain, A., Sarda, P., e Haritsa, J. R. (2004). Providing diversity in k-nearest neighbor query results. In *Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD)*, volume 3056 do *Lecture Notes in Computer Science*, páginas 404–413, Sydney, Australia. Springer, DOI: 10.1007/b97861.

[Jain e Dubes, 1988] Jain, A. K. e Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ.

[Jain et al., 1999] Jain, A. K., Murty, M. N., e Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys (CSUR)*, 31(3):264–323, DOI: 10.1145/331499.331504.

[Jin et al., 2004] Jin, W., Shi, R., e Chua, T.-S. (2004). A semi-naive bayesian method incorporating clustering with pair-wise constraints for auto image annotation. In *ACM International Conference on Multimedia (MULTIMEDIA)*, páginas 336–339, New York, NY, USA. ACM, DOI: 10.1145/1027527.1027605.

[Jin e French, 2005] Jin, X. e French, J. C. (2005). Improving image retrieval effectiveness via multiple queries. *Multimedia Tools and Applications*, 26(2):221–245, Kluwer, DOI: 10.1007/s11042-005-0453-5.

[Jing et al., 2003] Jing, F., Li, M., Zhang, L., Zhang, H.-J., e Zhang, B. (2003). Learning in region-based image retrieval. In *International Conference on Image and Video Retrieval (CIVR)*, volume 2728 do *Lecture Notes in Computer Science*, páginas 206–215, Urbana-Champaign, IL. Springer, DOI: 10.1007/3-540-45113-7_21.

[Jing e Baluja, 2008] Jing, Y. e Baluja, S. (2008). Visualrank: Applying pagerank to large-scale image search. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 30(11):1877–1890, IEEE, DOI: 10.1109/TPAMI.2008.121.

[Kaufman e Rousseeuw, 2005] Kaufman, L. e Rousseeuw, P. J. (2005). *Finding groups in data: An introduction to cluster analysis*. John Wiley and Sons.

[Kim e Chung, 2003] Kim, D.-H. e Chung, C.-W. (2003). Qcluster: relevance feedback using adaptive clustering for content-based image retrieval. In *International Conference*

on Management of Data (SIGMOD), páginas 599–610, San Diego, CA. ACM, DOI: 10.1145/872757.872829.

[Korn et al., 1996] Korn, F., Sidiropoulos, N., Faloutsos, C., Siegel, E., e Protopapas, Z. (1996). Fast nearest neighbor search in medical image databases. In *International Conference on Very Large Data Bases (VLDB)*, páginas 215–226, San Francisco, CA.

[Kotoulas e Andreadis, 2005] Kotoulas, L. e Andreadis, I. (2005). Real-time computation of zernike moments. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(6):801–809, DOI: 10.1109/TCSVT.2005.848302.

[Kruskal e Wish, 1978] Kruskal, J. B. e Wish, M. (1978). *Multidimensional Scaling*. SAGE Publications.

[Kuo et al., 2007] Kuo, C.-C., Glover, F., e Dhir, K. S. (2007). Analyzing and modeling the maximum diversity problem by zero-one programming. *Decision Sciences*, 24(6):1171–1185, DOI: 10.1111/j.1540-5915.1993.tb00509.x.

[Kutz et al., 2003] Kutz, O., Wolter, F., Sturm, H., Suzuki, N.-Y., e Zakharyaschev, M. (2003). Logics of metric spaces. *ACM Transactions on Computational Logic (TOCL)*, 4(2):260–294, ACM, DOI: 10.1145/635499.635504.

[Laguna e Marti, 1999] Laguna, M. e Marti, R. (1999). Grasp and path relinking for 2-layer straight line crossing minimization. *Informs Journal on Computing*, 11(1):44–52, Informs, DOI: 10.1287/ijoc.11.1.44.

[Lavrenko, 2009] Lavrenko, V. (2009). *A Generative Theory of Relevance (The Information Retrieval Series, vol. 26)*. Springer.

[Lee et al., 2006] Lee, H. D., Monard, M. C., e Wu, F. C. (2006). A fractal dimension based filter algorithm to select features for supervised learning. In *Advances in Artificial Intelligence (IBERAMIA-SBIA)*, volume 4140 do *Lecture Notes in Computer Science*, páginas 278–288, Ribeirão Preto, SP. Springer.

[Ley, 2009] Ley, M. (2009). dblp.xml – a documentation. DBLP Computer Science Bibliography, artigo e conjunto de dados disponíveis em http://dblp.uni-trier.de/xml/.

[Li et al., 2006] Li, S., Fevens, T., Krzyzak, A., e Li, S. (2006). Automatic clinical image segmentation using pathological modeling, PCA and SVM. *Engineering Applications of Artificial Intelligence*, (19):403–410, DOI: 10.1016/j.engappai.2006.01.011.

[Liu et al., 2006a] Liu, D., Hua, K. A., Vu, K., e Yu, N. (2006a). Efficient target search with relevance feedback for large cbir systems. In *ACM Symposium on Applied Computing (SAC)*, páginas 1393–1397, Dijon, França. ACM, DOI: 10.1145/1141277.1141598.

[Liu et al., 2006b] Liu, D., Hua, K. A., Vu, K., e Yu, N. (2006b). Fast query point movement techniques with relevance feedback for content-based image retrieval. In *International Conference on Extending Advances in Database Technology (EDBT)*, volume 3896 do *Lecture Notes in Computer Science*, páginas 700–717, Munich, Alemanha. Springer, DOI: 10.1007/11687238_42.

[Liu et al., 2007] Liu, Y., Zhang, D., Lu, G., e Ma, W.-Y. (2007). A survey of content-based image retrieval with high-level semantics. *Pattern Recognition*, 40(1):262–282, DOI: 10.1016/j.patcog.2006.04.045.

[LoBue e DeLoache, 2008] LoBue, V. e DeLoache, J. S. (2008). Detecting the snake in the grass: Attention to fear-relevant stimuli by adults and young children. *Psychological Science*, 19(3):284–289, Association for Psychological Science, DOI: 10.1111/j.1467-9280.2008.02081.x.

[Long et al., 2003] Long, F., Zhang, H., e Feng, D. D. (2003). *Fundamentals of Content-Based Image Retrieval (Multimedia Information Retrieval and Management - Technological Fundamentals and Applications)*. Springer.

[Longman Dictionary, 2003] Longman Dictionary (2003). Longman dictionary of contemporary english online. 4^a Edição, CDROM, Pearson Education.

[Luenberger, 1984] Luenberger, D. G. (1984). *Linear and Nonlinear Programming*. Addison-Wesley Inc., Reading, Massachusetts.

[MacArthur et al., 2000] MacArthur, S. D., Brodley, C. E., e Shyu, C.-R. (2000). Relevance feedback decision trees in content-based image retrieval. In *IEEE Workshop on Content-based Access of Image and Video Libraries*, páginas 68–72, Hilton Head Island, SC. IEEE, DOI: 10.1109/IVL.2000.853842.

[Manjunath et al., 2002] Manjunath, B. S., Salembier, P., e Sikora, T. (2002). *Introduction to MPEG-7: Multimedia Content Description Interface*. Wiley, New York.

[Manolopoulos et al., 2005] Manolopoulos, Y., Nanopoulos, A., Papadopoulos, A. N., e Theodoridis, Y. (2005). *R-Trees: Theory and Applications (Advanced Information and Knowledge Processing)*. Springer-Verlag, Secaucus, NJ, USA.

[Milanese e Cherbuliez, 1999] Milanese, R. e Cherbuliez, M. (1999). A rotation, translation, and scale-invariant approach to content-based image retrieval. *Visual Communication and Image Representation*, 10(2):186–196, DOI: 10.1006/jvci.1999.0411.

[Müller et al., 2004] Müller, H., Michoux, N., Bandon, D., e Geissbuhler, A. (2004). A review of content-based image retrieval systems in medical applications - clinical benefits

and future directions. *International Journal of Medical Informatics*, 73(1):1–23, DOI: 10.1016/j.ijmedinf.2003.11.024.

[Mokhtarian e Abbasi, 2002] Mokhtarian, F. e Abbasi, S. (2002). Shape similarity retrieval under affine transforms. *Pattern Recognition*, 35(1):31–41, DOI: 10.1016/S0031-3203(01)00040-1.

[Namnandorj et al., 2008] Namnandorj, S., Chen, H., Furuse, K., e Ohbo, N. (2008). Efficient bounds in finding aggregate nearest neighbors. In *International Conference on Database and Expert Systems Applications (DEXA)*, volume 5181 do *Lecture Notes in Computer Science*, páginas 693–700, Turin, Italia. Springer, DOI: 10.1007/978-3-540-85654-2_60.

[Nascimento e Toledo, 2008] Nascimento, M. C. V. e Toledo, F. M. B. (2008). A hybrid heuristic for the multi-plant capacitated lot sizing problem with setup carry-over. *Journal of the Brazilian Computer Society (JBCS)*, 14(4):7–15.

[Nelder e Mead, 1965] Nelder, J. A. e Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, 7:308–315.

[Pagel et al., 2000] Pagel, B.-U., Korn, F., e Faloutsos, C. (2000). Deflating the dimensionality curse using multiple fractal dimensions. In *International Conference on Data Engineering (ICDE)*, páginas 589–598, San Diego, California. DOI: 10.1109/ICDE.2000.839457.

[Papadias et al., 2005] Papadias, D., Tao, Y., Mouratidis, K., e Hui, C. K. (2005). Aggregate nearest neighbor queries in spatial databases. *ACM Transactions on Database Systems (TODS)*, 30(2):529–576, ACM, DOI: 10.1145/1071610.1071616.

[Pass et al., 1996] Pass, G., Zabih, R., e Miller, J. (1996). Comparing images using color coherence vectors. In *ACM International Conference on Multimedia (MULTIMEDIA)*, páginas 65–73, Boston, Massachusetts. ACM, DOI: 10.1145/244130.244148.

[Porkaew et al., 1999] Porkaew, K., Chakrabarti, K., e Mehrotra, S. (1999). Query refinement for multimedia similarity retrieval in MARS. In *ACM International Conference on Multimedia (MULTIMEDIA)*, páginas 235–238, Orlando, Florida. ACM, DOI: 10.1145/319463.319613.

[Prais e Ribeiro, 2000] Prais, M. e Ribeiro, C. C. (2000). Reactive grasp: An application to a matrix decomposition problem in tdma traffic assignment. *Informs Journal on Computing*, 12(3):164–176, Informs, DOI: 10.1287/ijoc.12.3.164.12639.

[Provost et al., 1998] Provost, F. J., Fawcett, T., e Kohavi, R. (1998). The case against accuracy estimation for comparing induction algorithms. In *International Conference on Machine Learning (ICML)*, páginas 445–453, San Francisco, CA. Morgan Kaufmann.

[Qamra et al., 2005] Qamra, A., Meng, Y., e Chang, E. Y. (2005). Enhanced perceptual distance functions and indexing for image replica recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 27(3):379–391, DOI: 10.1109/TPAMI.2005.54.

[Qian et al., 2002] Qian, W., Mao, F., Sun, X., Zhang, Y., Song, D., e Clarke, R. A. (2002). An improved method of region grouping for microcalcification detection in digital mammograms. *Computerized Medical Imaging and Graphics*, 26(6):361–368, DOI: 10.1016/S0895-6111(02)00045-9.

[Quinlan, 1993] Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers.

[Rahmani et al., 2005] Rahmani, R., Goldman, S. A., Zhang, H., Krettek, J., e Fritts, J. E. (2005). Localized content based image retrieval. In *ACM SIGMM International Workshop on Multimedia Information Retrieval (MIR)*, páginas 227–236, Hilton, Singapura. ACM, DOI: 10.1145/1101826.1101863.

[Razente et al., 2008a] Razente, H. L., Barioni, M. C. N., Traina, A. J. M., Faloutsos, C., e Traina-Jr., C. (2008a). A novel optimization approach to efficiently process aggregate similarity queries in metric access methods. In *ACM International Conference on Information and Knowledge Management (CIKM)*, páginas 193–202, Napa Valley, CA. DOI: 10.1145/1458082.1458110.

[Razente et al., 2008b] Razente, H. L., Barioni, M. C. N., Traina, A. J. M., e Traina-Jr, C. (2008b). Aggregate similarity queries in relevance feedback methods for content-based image retrieval. In *ACM Symposium on Applied Computing (SAC)*, páginas 869–874, Fortaleza, CE. DOI: 10.1145/1363686.1363887.

[Resende, 2009] Resende, M. G. (2009). Greedy randomized adaptive search procedures. In *Encyclopedia of Optimization*, páginas 1460–1469. Springer, DOI: 10.1007/978-0-387-74759-0_256.

[Resende e Ribeiro, 2003] Resende, M. G. C. e Ribeiro, C. C. (2003). Greedy randomized adaptive search procedures. *Handbook of Metaheuristics*, páginas 219–249, Kluwer, DOI: 10.1007/0-306-48056-5_8.

[Resende e Ribeiro, 2005] Resende, M. G. C. e Ribeiro, C. C. (2005). Grasp with path-relinking: Recent advances and applications. In *Metaheuristics: Progress as Real Problem Solvers*, páginas 29–63. Springer, DOI: 10.1007/0-387-25383-1_2.

[Ribeiro et al., 2002] Ribeiro, C. C., Uchoa, E., e Werneck, R. F. (2002). A hybrid grasp with perturbations for the steiner problem in graphs. *Informs Journal on Computing*, 14:228–246, DOI: 10.1287/ijoc.14.3.228.116.

[Ribeiro et al., 2005] Ribeiro, M. X., Balan, A. G. R., Felipe, J. C., Traina, A. J. M., e Traina-Jr, C. (2005). Mining statistical association rules to select the most relevant medical image features. In *IEEE International Workshop on Mining Complex Data (MCD)*, páginas 91–98, Houston, TX. IEEE.

[Rocchio, 1971] Rocchio, J. J. (1971). *Relevance Feedback in Information Retrieval*. The SMART Retrieval System: Experiments in Automatic Document Processing. Prentice-Hall, Englewood Cliffs, New Jersey.

[Roussopoulos et al., 1995] Roussopoulos, N., Kelley, S., e Vincent, F. (1995). Nearest neighbor queries. In *International Conference on Management of Data (SIGMOD)*, páginas 71–79, San Jose, CA. DOI: 10.1145/223784.223794.

[Rui et al., 1999] Rui, Y., Huang, T. S., e Chang, S.-F. (1999). Image retrieval: Current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation*, 10(1):39–62, DOI: 10.1006/jvci.1999.0413.

[Rui et al., 1997] Rui, Y., Huang, T. S., e Mehrotra, S. (1997). Content-based image retrieval with relevance feedback in mars. In *International Conference on Image Processing (ICIP)*, páginas 815–818, Washington, DC. IEEE, DOI: 10.1109/ICIP.1997.638621.

[Rui et al., 1998] Rui, Y., Huang, T. S., Ortega, M., e Mehrotra, S. (1998). Relevance feedback: A power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Video Technology*, 8(5):644–655, DOI: 10.1109/76.718510.

[Samet, 2003] Samet, H. (2003). Depth-first k-nearest neighbor finding using the maxnearestdist estimator. In *IEEE International Conference on Image Analysis and Processing (ICIAP)*, páginas 486–491, Mantova, Itália. DOI: 10.1109/ICIAP.2003.1234097.

[Samet, 2006] Samet, H. (2006). *Foundations of Multidimensional and Metric Data Structures*. Morgan Kaufmann, San Francisco, CA.

[Santini e Gupta, 2001] Santini, S. e Gupta, A. (2001). A wavelet data model for image databases. In *IEEE International Conference on Multimedia and Expo (ICME)*, páginas 345–348, Tóquio, Japão. IEEE, DOI: 10.1109/ICME.2001.1237919.

[Santini e Jain, 1996] Santini, S. e Jain, R. (1996). Gabor space and the development of preattentive similarity. In *International Conference on Pattern Recognition*, páginas 40–44, Vienna, Austria. DOI: 10.1109/ICPR.1996.545988.

[Schilham et al., 2006] Schilham, A. M. R., van Ginneken, B., e Loog, M. (2006). A computer-aided diagnosis system for detection of lung nodules in chest radiographs with an evaluation on a public database. *Medical Image Analysis*, 10(2):247–258, DOI: 10.1016/j.media.2005.09.003.

[Seidl e Kriegel, 1998] Seidl, T. e Kriegel, H.-P. (1998). Optimal multi-step k-nearest neighbor search. In *International Conference on Management of Data (SIGMOD)*, páginas 154–165, Seattle, Washington. DOI: 10.1145/276304.276319.

[Sellis et al., 1987] Sellis, T. K., Roussopoulos, N., e Faloutsos, C. (1987). The r+-tree: A dynamic index for multi-dimensional objects. In *International Conference on Very Large Data Bases (VLDB)*, páginas 507–518, Brighton, Inglaterra.

[Shen et al., 2009] Shen, H. T., Jiang, S., Tan, K.-L., Huang, Z., e Zhou, X. (2009). Speed up interactive image retrieval. *International Journal on Very Large Data Bases (VLDB Journal)*, 18(1):329–343, Springer-Verlag, DOI: 10.1007/s00778-008-0101-6.

[Shi e Malik, 2000] Shi, J. e Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22(8):888–905, DOI: 10.1109/34.868688.

[Shi et al., 2004] Shi, R., Feng, H., Chua, T.-S., e Lee, C.-H. (2004). An adaptive image content representation and segmentation approach to automatic image annotation. In *International Conference on Image and Video Retrieval (CIVR)*, volume 3115 do *Lecture Notes in Computer Science*, páginas 545–554, Dublin, Irlanda. Springer, DOI: 10.1007/b98923.

[Siadat e Soltanian-Zadeh, 2005] Siadat, M.-R. e Soltanian-Zadeh, H. (2005). Content-based image database system for epilepsy. *Computer Methods and Programs in Biomedicine*, (79):209–226, DOI: 10.1016/j.cmpb.2005.03.012.

[Silva et al., 2007] Silva, G. C., de Andrade, M. R. Q., Ochi, L. S., Martins, S. L., e Plastino, A. (2007). New heuristics for the maximum diversity problem. *Journal of Heuristics*, 13(4):315–336, DOI: 10.1007/s10732-007-9010-x.

[Smeulders et al., 2000] Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., e Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22(12):1349–1380, IEEE Computer Society, DOI: 10.1109/34.895972.

[Soltanian-Zadeh et al., 2004] Soltanian-Zadeh, H., Rafiee-Rad, F., e Pourabdollah-Nejad, S. (2004). Comparison of multiwavelet, wavelet, haralick, and shape features for microcalcification classification in mammograms. *Pattern Recognition*, (37):1973–1986, DOI: 10.1016/j.patcog.2003.03.001.

[Sousa et al., 2002] Sousa, E. P., Traina-Jr, C., Traina, A. J. M., e Faloutsos, C. (2002). How to use fractal dimension to find correlations between attributes. In *ACM SIGKDD Workshop on Fractals and Self-similarity in Data Mining: Issues and Approaches*, páginas 26–30, Edmonton, Canadá.

[Stehling et al., 2003] Stehling, R. O., Nascimento, M. A., e Falcão, A. X. (2003). Cell histograms versus color histograms for image representation and retrieval. *Knowledge and Information Systems*, 5(3):315–336, Springer-Verlag, DOI: 10.1007/s10115-003-0084-y.

[Sun et al., 2006] Sun, J., Zhang, X., Cui, J., e Zhou, L. (2006). Image retrieval based on color distribution entropy. *Pattern Recognition Letters*, 27(10):1122–1126, DOI: 10.1016/j.patrec.2005.12.014.

[Swain e Ballard, 1991] Swain, M. J. e Ballard, D. H. (1991). Color indexing. *International Journal of Computer Vision*, 7(1):11–32, Springer, DOI: 10.1007/BF00130487.

[Tahaghoghi et al., 2002] Tahaghoghi, S. M. M., Thom, J. A., e Williams, H. E. (2002). Multiple example queries in content-based image retrieval. In *International Symposium on String Processing and Information Retrieval (SPIRE)*, volume 2476 do *Lecture Notes in Computer Science*, páginas 227–241, Lisboa, Portugal. Springer, DOI: 10.1007/3-540-45735-6_20.

[Tasan e Ozsoyoglu, 2004] Tasan, M. e Ozsoyoglu, Z. M. (2004). Improvements in distance-based indexing. In *International Conference on Scientific and Statistical Database Management (SSDBM)*, páginas 161–170, Santorini, Grécia. IEEE, DOI: 10.1109/SSDM.2004.1311208.

[Traina et al., 2006] Traina, A. J. M., Marques, J., e Traina-Jr, C. (2006). Fighting the semantic gap on CBIR systems through new relevance feedback techniques. In *IEEE Symposium on Computer-Based Medical Systems (CBMS)*, páginas 881–886, Salt Lake City, Utah. IEEE, DOI: 10.1109/CBMS.2006.88.

[Traina et al., 2003] Traina, A. J. M., Traina-Jr, C., Bueno, J. M., Chino, F. J. T., e Marques, P. M. A. (2003). Efficient content-based image retrieval through metric histograms. *World Wide Web Journal*, 6(2):157–185, Kluwer, DOI: 10.1023/A:1023670521530.

[Traina-Jr et al., 2002] Traina-Jr, C., Traina, A. J. M., Faloutsos, C., e Seeger, B. (2002). Fast indexing and visualization of metric datasets using Slim-trees. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 14(2):244–260, DOI: 10.1109/69.991715.

[Traina-Jr et al., 2000] Traina-Jr, C., Traina, A. J. M., Wu, L., e Faloutsos, C. (2000). Fast feature selection using fractal dimension. In *Simpósio Brasileiro de Banco de Dados (SBBD)*, páginas 158–171, João Pessoa, PB.

[Vailaya et al., 2001] Vailaya, A., Figueiredo, M. A. T., Jain, A. K., e Zhang, H.-J. (2001). Image classification for content-based indexing. *IEEE Transactions on Image Processing*, 10(1):117–130, DOI: 10.1109/83.892448.

[van Leuken et al., 2009] van Leuken, R. H., Garcia, L., Olivares, X., e van Zwol, R. (2009). Visual diversification of image search results. In *International Conference on World Wide Web (WWW)*, páginas 341–350, Madrid, Espanha. ACM, DOI: 10.1145/1526709.1526756.

[van Zwol et al., 2008] van Zwol, R., Murdock, V., Pueyo, L. G., e Ramirez, G. (2008). Diversifying image search with user generated content. In *ACM SIGMM International Conference on Multimedia Information Retrieval (MIR)*, páginas 67–74, Vancouver, British Columbia, Canada. ACM, DOI: 10.1145/1460096.1460109.

[Vee et al., 2008] Vee, E., Srivastava, U., Shanmugasundaram, J., Bhat, P., e Amer-Yahia, S. (2008). Efficient computation of diverse query results. In *International Conference on Data Engineering (ICDE)*, páginas 228–236, Cancún, México. DOI: 10.1109/ICDE.2008.4497431.

[Vieira et al., 2006] Vieira, M. R., Traina-Jr, C., Chino, F. J. T., e Traina, A. J. M. (2006). Dbm-tree: Trading height-balancing for performance in metric access methods. *Journal of the Brazilian Computer Society (JBCS)*, 11(3):37–52.

[Vieira et al., 2007] Vieira, M. R., Traina-Jr, C., Traina, A. J. M., Arantes, A. S., e Faloutsos, C. (2007). Boosting k-nearest neighbor queries estimating suitable query radii. In *International Conference on Scientific and Statistical Database Management (SSDBM)*, páginas 1–10, Banff, Canadá. DOI: 10.1109/SSDBM.2007.5.

[Wang et al., 1999] Wang, J. T.-L., Wang, X., Lin, K.-I., Shasha, D., Shapiro, B. A., e Zhang, K. (1999). Evaluating a class of distance-mapping algorithms for data mining and clustering. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, páginas 307–311, San Diego, California. ACM, DOI: 10.1145/312129.312264.

[Wei et al., 2009] Wei, L., Yang, Y., e Nishikawa, R. M. (2009). Microcalcification classification assisted by content-based image retrieval for breast cancer diagnosis. *Pattern Recognition*, 42(6):1126–1132, Elsevier Science Inc., DOI: 10.1016/j.patcog.2008.08.028.

[Wright, 1929] Wright, W. D. (1929). A re-determination of the trichromatic coefficients of the spectral colours. *Transactions of the Optical Society*, 30(4):141–164.

[Wu et al., 2000] Wu, L., Faloutsos, C., Sycara, K., e Payne, T. R. (2000). Falcon: Feedback adaptive loop for content-based retrieval. In *International Conference on Very Large Databases (VLDB)*, páginas 297–306, Cairo, Egito.

[Wu e Manjunath, 2001] Wu, P. e Manjunath, B. S. (2001). Adaptive nearest neighbor search for relevance feedback in large image databases. In *ACM International Conference on Multimedia (MULTIMEDIA)*, páginas 89–97, Ottawa, Canadá. ACM, DOI: 10.1145/500141.500157.

[Xin et al., 2006] Xin, D., Cheng, H., Yan, X., e Han, J. (2006). Extracting redundancy-aware top-k patterns. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, páginas 444–453, Philadelphia, PA. DOI: 10.1145/1150402.1150452.

[Yianilos, 1993] Yianilos, P. N. (1993). Data structures and algorithms for nearest neighbor search in general metric spaces. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, páginas 311–321, Austin, TX.

[Yin et al., 2005] Yin, P.-Y., Bhanu, B., Chang, K.-C., e Dong, A. (2005). Integrating relevance feedback techniques for image retrieval using reinforcement learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 27(10):1536–1551, DOI: 10.1109/TPAMI.2005.201.

[Zezula et al., 2006] Zezula, P., Amato, G., Dohnal, V., e Batko, M. (2006). *Similarity Search: The Metric Space Approach (Series Advances in Database Systems, vol. 32)*. Springer.

[Zhang et al., 2005] Zhang, C., Chai, J. Y., e Jin, R. (2005). User term feedback in interactive text-based image retrieval. In *ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, páginas 51–58, Salvador (BA). ACM, DOI: 10.1145/1076034.1076046.

[Zhang et al., 2006] Zhang, J., Zhou, X., Wang, W., Shi, B., e Pei, J. (2006). Using high dimensional indexes to support relevance feedback based interactive images retrieval. In *International Conference on Very Large Data Bases (VLDB)*, páginas 1211–1214, Seoul, Korea. VLDB Endowment.

[Zhou et al., 2005] Zhou, Q., Ma, L., Celenk, M., e Chelberg, D. M. (2005). Content-based image retrieval based on ROI detection and relevance feedback. *Multimedia Tools and Applications*, 27(2):251–281, DOI: 10.1007/s11042-005-2577-z.

[Zhou e Huang, 2003] Zhou, X. S. e Huang, T. S. (2003). Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems*, 8(6):536–544, DOI: 10.1007/s00530-002-0070-3.

[Zhou et al., 2006] Zhou, Z.-H., Chen, K.-J., e Dai, H.-B. (2006). Enhancing relevance feedback in image retrieval using unlabeled data. *ACM Transactions on Information Systems (TOIS)*, 24(2):219–244, ACM, DOI: 10.1145/1148020.1148023.

As referências bibliográficas que apresentam o identificador *DOI* (*Digital Object Identifier*) podem ser acessadas por meio do endereço http://dx.doi.org/ acrescidas do identificador no endereço. Informações sobre o sistema *DOI* podem ser obtidas em http://www.doi.org/.