



Universidade Federal do Amazonas
Instituto de Computação
Programa de Pós-Graduação em Informática

DESCRITOR DE IMAGENS ADAPTÁVEL A DIFERENTES DOMÍNIOS

Márcio Luiz Assis Vidal

Manaus – Amazonas
Março de 2013

Márcio Luiz Assis Vidal

DESCRITOR DE IMAGENS ADAPTÁVEL A DIFERENTES DOMÍNIOS

Tese apresentada ao Programa de Pós-Graduação em Informática da Universidade Federal do Amazonas, como requisito parcial para a obtenção do grau de Doutor em Informática.

Orientador: Prof. João Marcos Bastos Cavalcanti, Ph.D.

Márcio Luiz Assis Vidal

DESCRITOR DE IMAGENS ADAPTÁVEL A DIFERENTES DOMÍNIOS

Tese apresentada ao Programa de Pós-Graduação em Informática da Universidade Federal do Amazonas, como requisito parcial para a obtenção do grau de Doutor em Informática.

Banca Examinadora

Prof. João Marcos Bastos Cavalcanti, Ph.D. – Orientador
Instituto de Computação – UFAM

Prof. Dr. Edleno Silva de Moura
Instituto de Computação – UFAM

Prof. Fulano de tal, Ph.D.
Departamento de BLA

Prof. Fulano de tal, Ph.D.
Departamento de BLA

Manaus – Amazonas
Dezembro de 2012

A todos que me acompanharam nessa trajetória.

Agradecimentos

Lorem ipsum dolor sit amet, integre lucilius vituperata no sea, nonumes probatus ea quo. Ne quo dicta consulatu. Sea modus tritani imperdiet ut, mei ne veri assueverit, erat novum quo te. Sea no tantas labore. Denique consequuntur quo eu. Quem nemore cum ad. Ut his prompta evertitur definitiones.

Te purto ferri perfecto quo, vix legimus appellantur ea, quodsi intellegebat vel te. Te pri odio nostrum, agam eros nostro ne eum, has ei essent diceret mnesarchum. Te malorum mentitum sea, et choro disputando quo. Argumentum omittantur no eum.

Zril equidem pro cu. Nam falli zril intellegat et, no mel duis ullamcorper. At putant interesset eam, ea erat laboramus dissentiunt ius, quem tantas ex quo. Probo repudiare persequeris ad eam, eos no iuvaret offendit eleifend. Per omnes mediocrem potentium id, no malis ullamcorper eum. Sea ea viris liberavisse, omnes democritum persequeris ad mel, ea quot dicit concludaturque pri. Vix in probo mundi quidam.

Ea nec animal feugiat, ea vide postea invidunt per. Nec eligendi appellantur ex, ne usu graeco deleniti corrumpit. Mea iisque aeterno potentium ad, cu vim feugait dolores appellantur. Ei nam labore invidunt, tota corpora liberavisse at nec. Esse quas lorem ea qui, oporteat dissentiunt in est, quem altera eum id. Eos vide clita ex, autem aperiri cu vis, eu duo facilis conclusionemque.

Probo appellantur cu vel. Ludus numquam per eu. Ad dolor persius dignissim sed, ne his nobis facilisi. Cibo denique at mei, mea magna posse no. Pri dolorem evertitur ei, ea veritus efficiendi vim, quo ad error homero contentiones.

*Subi em um pé de laranja para apanhar
jabuticaba, mas como estava no tempo de
alface roubaram minha bicicleta!*

Resumo

Lorem ipsum dolor sit amet, integre lucilius vituperata no sea, nonumes probatus ea quo. Ne quo dicta consulatu. Sea modus tritani imperdiet ut, mei ne veri assueverit, erat novum quo te. Sea no tantas labore. Denique consequuntur quo eu. Quem nemore cum ad. Ut his prompta evertitur definitiones.

Te purto ferri perfecto quo, vix legimus appellantur ea, quodsi intellegebat vel te. Te pri odio nostrum, agam eros nostro ne eum, has ei essent diceret mnesarchum. Te malorum mentitum sea, et choro disputando quo. Argumentum omittantur no eum. Zril equidem pro cu. Nam falli zril intellegat et, no mel duis ullamcorper. At putant interesseret eam, ea erat laboramus dissentiunt ius, quem tantas ex quo. Probo repudiare persequeris ad eam, eos no iuvaret offendit eleifend. Per omnes mediocrem potentium id, no malis ullamcorper eum. Sea ea viris liberavisse, omnes democritum persequeris ad mel, ea quot dicit concludaturque pri. Vix in probo mundi quidam.

Ea nec animal feugiat, ea vide postea invidunt per. Nec eligendi appellantur ex, ne usu graeco deleniti corrumpit. Mea iisque aeterno potentium ad, cu vim feugait dolores appellantur. Ei nam labore invidunt, tota corpora liberavisse at nec. Esse quas lorem ea qui, oporteat dissentiunt in est, quem altera eum id. Eos vide clita ex, autem aperiri cu vis, eu duo facilis conclusionemque. Probo appellantur cu vel. Ludus numquam per eu. Ad dolor persius dignissim sed, ne his nobis facilisi. Cibo denique at mei, mea magna posse no. Pri dolorem evertitur ei, ea veritus efficiendi vim, quo ad error homero contentiones.

Conteúdo

Lista de Figuras	ii
1 Introdução	1
1.1 Motivação	1
1.2 Objetivos	2
1.3 Contribuições	2
1.4 Publicação	3
1.5 Organização da Tese	3
2 Content Based Image Retrieval - CBIR	5
3 Trabalhos relacionados	9
3.1 Descritores de Cor	10
3.2 Descritores de Textura	13
3.3 Descritores de Forma	15
3.4 Bag of Visual Words/Features	17
3.5 Modelo Vetorial	18
4 SDLF - Sorted Dominant Local Feature	21
4.1 Detalhamento dos passos do método proposto	24
5 Experimentos	27
5.1 Bases de imagens	28
5.2 Experimentos de configuração do SDLF	31
5.3 Experimentos	33
5.4 Discussão	37
6 Conclusões	39

Lista de Figuras

2.1	Visão geral do processo de busca - CBIR	6
3.1	Imagens diferentes com mesmo histograma de cor ([24])	10
4.1	<i>Exemplo de extração do código textual.</i>	23
4.2	<i>Exemplo de extração do código textual.</i>	24
4.3	Bloco na região A2	25
4.4	Bloco na região A5	25
5.1	Consultas utilizadas para base WANG.	29
5.2	Consultas utilizadas para base CCD.	30
5.3	Escolha do número de blocos e limiar.	32
5.4	Blocos se adaptam às dimensões da imagem.	32
5.5	Consultas utilizadas para base ROUPAS.	33
5.6	Resultados obtidos com a base <i>ROUPAS</i>	35
5.7	Resultados obtidos com a base <i>ROUPAS</i>	36
5.8	Resultados obtidos com a base <i>CCD</i>	36
5.9	Resultados obtidos com a base <i>WANG</i>	37
5.10	Falha no método para o caso do descritor baseado exclusivamente nas propriedades de cor das imagens.	38

Capítulo 1

Introdução

A evolução e popularização dos dispositivos de captura de imagens digital, aliados à crescente difusão das redes sociais, resultou em uma notável explosão no volume de imagens digitais na Internet. Como consequência deste vertiginoso e desordenado crescimento surgiu a necessidade de classificar e/ou recuperar imagens relacionadas a um determinado assunto ou a uma imagem específica de forma eficiente e eficaz.

1.1 Motivação

No intuito de resolver o problema de se classificar e/ou recuperar imagens relacionadas a um determinado assunto ou a uma imagem específica, muitos métodos tem sido propostos. Inicialmente tais métodos utilizavam a informação textual para descrever as imagens, como as técnicas propostas por [4, 5], utilizando esta informação textual associada às de técnicas de Recuperação de Informação bem consolidadas como o modelo vetorial proposto por [30], era realizada a recuperação de imagens. Porém devido a existência de bases de imagens cujas informações textuais são insuficientes ou inexistentes (bases de imagens de sensoriamento remoto, imagens fotográficas e etc.), técnicas de busca de imagens baseadas somente em conteúdo visual (Content Based Image Retrieval, de agora em diante chamada de CBIR) foram propostas, assim como as técnicas que combinam ambas as propriedades (textual e o conteúdo da imagem), como as propostas por [23] e [8].

A recuperação de imagens baseada em conteúdo depende de vários fatores como a escolha de quais propriedades da imgem como cor, forma, textura, regiões etc, serão usadas para descrever a imagem; qual o método de extração destas propriedades; medida de similaridade entre imagens; qual o método de indexação a ser usado; etc. Desta forma, a melhoria de qualquer destes fatores ou, em alguns casos, a combinação destes podem melhorar a eficácia da recuperação das imagens baseada em conteúdo.

Em todos os casos verificados sempre existe uma lacuna entre a eficiência e eficácia, devido ao custo computacional, seja durante o processo de extração de propriedades ou no processo de cálculo de similaridade, por esta razão o trabalho aqui proposto procura a eficiência dos descritores a robustez, confiança e velocidade das técnicas de *Recuperação de Informação - RI*.

1.2 Objetivos

O objetivo deste trabalho é propor um descritor de imagens que seja adaptável a diferentes tarefas de busca e classificação, capaz de permitir ajustes conforme as propriedades mais relevantes de um domínio de aplicação selecionado.

Este descritor deve ser facilmente acoplado a técnicas de *RI* amplamente conhecidas para ser utilizado no processo de CBIR. Porém, para que isso ocorra a contento, temos que responder algumas questões. 1. Como descrever as imagens de forma que cada imagem possa ser vista como uma codificação textual que será utilizado pelas técnicas de *RI* tradicionais? 2. Que propriedades da imagem melhor caracterizam determinado domínio de aplicação?. As respostas para essas perguntas podem ser encontradas no Capítulo 4.

1.3 Contribuições

As contribuições originais deste trabalho podem ser elencados como segue:

- Criação e implementação de um descritor de imagens;
- Estudo e validação das propriedades das imagens que melhor descrevem os domínios de aplicação *WEB* e comércio eletrônico;
- Combinação entre as áreas de *CBIR* e *RI*.

1.4 Publicação

Com o objetivo de validar o descritor proposto nesta Tese de Doutorado, perante a comunidade científica, publicamos este trabalho na 21st International Conference on Pattern Recognition - ICPR2012. Com o título 'Sorted Dominant Local Color for Searching Large and Heterogeneous Image Databases'.

1.5 Organização da Tese

Esta Tese está organizada como segue: Capítulo 2, referente aos conceitos de *CBIR*; no Capítulo 3 é feito um levantamento bibliográfico referente ao trabalho proposto; o descritor proposto é apresentado no Capítulo 4 desta Tese; Os detalhes dos experimentos, bem como os experimentos de setup do descritor estão detalhados no Capítulo 5 e no Capítulo 6 encerramos com as nossas conclusões.

Capítulo 2

Content Based Image Retrieval - CBIR

A Idéia básica por trás da CBIR é construir uma representação de características de imagens, que consiste em extrair as propriedades das imagens de acordo com um critério específico (cor, textura, regiões, etc.) ou com a combinação destes. O resultado do processo de extração é um vetor de características que codifica as propriedades da imagem. Após o processo de extração é realizada a indexação dos vetores de características. Quando uma imagem de consulta é passada durante o processo de busca, suas propriedades também são extraídas de acordo com os mecanismos de extração utilizados para a criação do vetor de características. O resultado do processo de busca é baseado na similaridade entre o vetor de característica da imagem de consulta e os vetores de características do banco de dados e os seus resultados são classificados de acordo com algum critério de similaridade. A Figura 2.1 apresenta uma visão geral deste processo.

Como mencionado anteriormente o processo de CBIR consiste de três fases básicas: Extração de propriedades da imagem; Indexação e o Cálculo de similaridade.

A **Extração de propriedades da imagem** consiste em descrever a imagem por

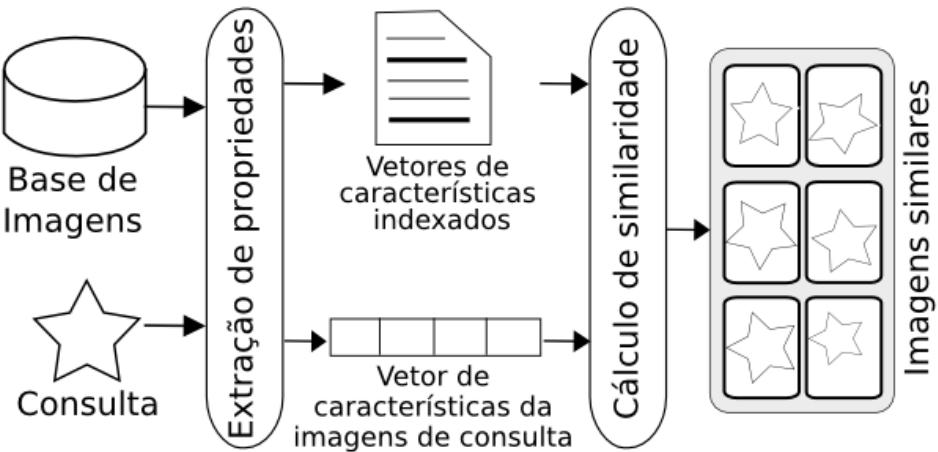


Figura 2.1: Visão geral do processo de busca - CBIR

meio de vetores de características onde cada elemento destes vetores representa as propriedades da imagem obtidas a partir da utilização de algum método descritor de imagem. Atualmente existe uma grande variedade de métodos na literatura, estes métodos são categorizados conforme as propriedades utilizadas para descrever a imagem como cor, textura, divisão em regiões, etc.

A etapa de **indexação** é de fundamental importância dado o atual volume imagens disponíveis na Web e seu contínuo crescimento, a ausência desta etapa torna, em alguns casos, o processo de busca inviável, principalmente para os casos onde o tempo de resposta deve ser considerado, como o ambiente Web.

Uma vez que as imagens sejam descritas em forma de vetores, e estes vetores de carcterísticas sejam indexados, pode-se iniciar o processo de busca. Este processo, consiste em utilizar o mesmo método de descrição de imagem usado para a criação da base, e utilizar alguma métrica para calcular a **similaridade** entre os vetores de consulta e os vetores da base. O resultado obtido é uma lista de imagens ordenadas conforme o resultado da métrica adotada para o calculo de similaridade entre a imagem de consulta e as imagens indexadas.

A idéia básica desta proposta baseia-se na premissa de que o processo de busca de imagens é intimamente dependente das propriedades da imagens como cor, textura, forma, etc., bem como do seu domínio de aplicação, que pode ser dividido em:

domínio geral com bases grandes e heterogêneas (Web) e domínio específico como bases de imagens de raio X, imagens de impressões digitais, sensoriamento remoto, comércio eletrônico, etc. As imagens de cada um destes domínios de aplicação possuem propriedades particulares que devem ser exploradas pelo processo de CBIR direcionado ao domínio de aplicação em questão.

A escolha do método a ser adotado não é uma tarefa trivial, visto que é comum observarmos na literatura trabalhos onde os experimentos são realizados em bases relativamente homogêneas e com um pequeno número de imagens, como visto nos trabalhos de [40], [36], [9], [34], [39] e [10]. O fato de não se usar uma base heterogênea e com um número significativo de imagens pode mascarar o real desempenho do descritor.

No trabalho proposto por [17] bem como no experimentos realizados para validar o método proposto neste trabalho, onde os experimentos foram conduzidos em bases com mais de 100.000 imagens, foi observado que em bases grandes e heterogêneas os resultados obtidos pelos métodos testados tiveram grande variação de resultados, enquanto que os mesmos métodos testados em bases pequenas, como a base *Wang* proposta por [19] e *CCD* proposta por [7], tiveram resultados próximos com diferenças estatisticamente insignificante. Esse resultado ratifica a idéia de que realizar experimentos, com foco em CBIR genérica, em bases com um número reduzido de imagens pode mascarar o desempenho do descritor.

Outro fator de grande influência na escolha do método a ser adotado, em sistemas de CBIR onde o desempenho é um fator crucial como sistemas Web, é o tempo de extração das propriedades das imagens. Neste caso deve-se adotar mecanismos de baixo custo computacional. Porém tal escolha pode afetar diretamente sua eficácia, desta forma é imperativo que o método escolhido alcance o equilíbrio entre eficiência e eficácia. Percebe-se então que encontrar quais propriedades das imagens resultam em descritor com melhor equilíbrio entre eficiência e eficácia é uma investigação a ser feita.

Capítulo 3

Trabalhos relacionados

Os métodos propostos para CBIR baseiam-se em algoritmos de extração de propriedades e medidas de similaridade das imagens. Estes métodos são classificados de acordo com as propriedades utilizadas para descrever a imagem. Tais propriedades são a cor, textura e forma utilizadas durante o processo de representação da imagem ([36]). É importante observar que cada método proposto tem suas vantagens e desvantagens e a escolha de um deles depende principalmente do seu domínio de aplicação. Por exemplo, a utilização de métodos com alto custo computacional como os descritores de textura não devem ser utilizadas em sistemas voltados para Web onde o tempo de resposta é tão importante quanto a precisão da resposta obtidas.

Note que para qualquer classe de descritores adotada, o descritor extrai as propriedades da imagem de maneira *geral* (gera um único vetor de características para imagem como um todo), através do particionamento *fixo* ou *segmentação automática* da imagem (para cada imagem é gerado um vetor de característica para cada partição ou segmento da imagem) ou *local* onde se extrai as propriedades dos pontos de interesse na imagem e os caracterizam por informações invariantes a transformações como orientação e escala. É fácil observar que apesar da abordagem de extração global ser computacionalmente mais barata, imagens completamente diferentes podem possuir histogramas iguais, como visto na Figura 3.1, este problema é facilmente

resolvido utilizando a abordagem de particionamento/segmentação da imagem. A seguir comentaremos cada uma destas classes de descritores e apresentaremos os trabalhos de CBIR diretamente relacionados à nossa proposta que utilizam a abordagem de extração de propriedades locais da imagem.



Figura 3.1: Imagens diferentes com mesmo histograma de cor ([24])

Como mencionado acima, a escolha do método é baseada no domínio de aplicação onde este será usado, contudo determinar qual o melhor método, dentre os existentes, é uma tarefa extremamente difícil e não delimitada no escopo deste trabalho.

Em [26] foi realizado um estudo comparativo com 88 descritores existentes na literatura, neste trabalho foi observado que quanto maior o número de imagens na base e quanto maior sua heterogeneidade, os descritores mais indicados a tarefa de busca são os descritores de cor.

3.1 Descritores de Cor

Descritores baseados no histograma de cor, que indica a frequência da ocorrência de cada cor em uma determinada imagem. Amplamente utilizados para CBIR em aplicações web, dada sua eficiência e insensibilidade às pequenas mudanças no ponto de vista da câmera, como discutidos por ([31]).

Em [39, 40], é proposta uma maneira de representar a informação de cor e estrutura espacial com um único descritor chamado de Dominant Color Structure Descriptor (DCSD). Este descritor combina a compactação do Dominant Color Descriptor (histograma global) e a precisão do Color Structure Descriptor (histograma local), tratando suas falhas para melhorar significativamente a performance do des-

citor proposto, este descritor é composto por duas fases chamadas de *Quantização de cor* e *Escaneamento da estrutura do bloco*. A etapa de *Quantização de cor* utiliza o número de clusters gerados pelo GLA (Generalized Lloyd Algorithm) [12] para especificar a quantidade de cores quantizadas da imagem, então para cada pixel da imagem é verificada a distância entre a cor original e a cor dominante quantizada segundo um threshold de 20. Para os casos onde o pixel não casa com nenhuma das cores quantizadas, este pixel é desconsiderado.

A etapa de *Escaneamento da estrutura do bloco* gera os bins do histograma pela soma dos elementos dentro da estrutura 8x8 pixels. Como medida de similaridade foi utilizada a distância Euclidiana. O método utiliza a Averaged Normalized Modified Retrieval Rate (ANMRR) como métrica de avaliação, utilizando este método de avaliação o trabalho proposto foi 18,54% mais preciso que o segundo colocado o CSD. Porém a base utilizada foi a Common Color Dataset (CCD) que contém 5466 images, uma base pequena e com suas imagens visualmente muito próximas. É importante observar que nestes trabalhos não foi sugerida a utilização de avaliadores para determinar o grau de similaridade entre a imagem data como consulta e as imagens retornadas pelo método.

[32], propôs uma padronização de descritores de streaming de vídeo/imagens ou vídeos/imagens armazenadas ou bits do cabeçalho de vídeo padronizados (Descritores visuais de baixo nível) que ajudam os usuários ou aplicativos a identificar, classificar e filtrar imagens ou vídeo o padrão MPEG-7. Estes descritores de baixo nível pode ser usado para comparar, filtrar, ou procurar imagens ou vídeos puramente baseado em descrições não textuais da imagem. Devido às suas características descritivas, o desafio para o desenvolvimento de tais descritores MPEG-7 não textuais é que eles devem ser significativos no contexto de várias aplicações.

Dentre os descritores do padrão MPEG-7 o Color Structure Descriptor - CSD foi o que obteve os melhores resultados no experimentos de comparação de descritores realizados por [26]. O CSD usa um elemento estruturante de 8x8 pixels para percor-

rer a imagem, sendo que esta deve estar no espaço de cor HMMD (hue, max, min, diff). À medida que o elemento estruturante percorre a imagem, um histograma é atualizado. Cada bin $h(m)$ do histograma é incrementado sempre que a cor m estiver dentro da janela, onde m varia de 0 a $M - 1$ e M indica a quantização do espaço de cor. O elemento estruturante pode ter amostragens diferentes dependendo do tamanho da imagem. Este tamanho de amostragem também indica qual o passo do deslocamento em pixels do elemento estruturante durante a varredura da imagem. Ao final, cada bin $h(m)$ do histograma representa a quantidade de locais em que a janela esteve na imagem nos quais havia um pixel de cor m dentro dela. Após a imagem ser totalmente percorrida o histograma é normalizado pela quantidade de locais em que o elemento estruturante fica durante o percurso. Os valores de cada bin são então quantizados não linearmente em 8 bits por bin. Como medida de similaridade é usada a medida L1.

Ainda no contexto do padrão MPEG-7 existe o *Color Layout Descriptor - CLD*, este descriptor representa a distribuição espacial de propriedades cores em uma imagem. Esta distribuição espacial de cores é obtida aplicando a transformada discreta de cosseno no espaço bidimensional. A imagem é dividida em 8×8 blocos não sobrepostos e para cada bloco é determinada sua representação de cor. Adotando o espaço quantizado de cor YCbCr para o padrão CLD.

[9] abordam a ideia de particionamento fixo da imagem, este processo é realizado em duas etapas. Primeiro cada imagem é particionada em 16 partes (4×4 blocos), depois disso é feita a segmentação do background/foreground para cada bloco. O princípio básico desta segmentação é que para os valores baixos de saturação um pixel pode ser aproximado pelo seu valor, enquanto para valores altos o pixel pode ser aproximado pela matiz, esta aproximação é feita utilizando um threshold empírico para cada bloco de acordo com a matiz das regiões de valores dominantes. A partir desta decomposição, características de baixo nível da imagem podem ser extraídas de cada uma destas regiões disjuntas para cada blocos em cada imagem. Para

isto foi usado o histograma de cor quantizado pro HSV, devido à sua simplicidade. Como métrica de similaridade foi usada a disntância Euclidiana. Como base para os experimentos deste método, foram selecionadas 500 imagens heterogêneas da base do COREL, uma base extremamente pequena que não impõe dificuldades para qualquer método. O resultado dos experimentos foram medidos utilizando medidas de precisão/revocação, que não se aplicam a buscas base grandes e que não se tem definido o pool de respostas relevantes, como a Web. Nota-se neste trabalho que não foi sugerida a utilização de avaliadores para determinara o grau de similaridade entre a imagem data como consulta e as imagens retornadas pelo método para o cálculo da precisão/revocação.

O método Border and Interior Classification - BIC, proposto por [33], classifica os pixels da imagem em pixels de borda ou pixels de interior. Seu algoritmo de extração inicialmente quantiza a imagem em 64 cores. Os pixels são então classificados pela seguinte regra: se um pixel possui a mesma cor dos seus vizinhos-4 (acima, à direita, abaixo e à esquerda) ele é classificado como interior; caso contrário ele é classificado como borda. Ao final, são calculados dois histogramas de cor, sendo um para os pixels de borda e outro para os pixels de interior. Os dois histogramas são concatenados e armazenados como um único histograma de 128 bins. A comparação dos histogramas é feita utilizando-se a distância dLog. Em [26] e [27] foi demonstrado que apesar de possuir um algoritmo simples este método obteve uma elevada eficácia em bases de imagens heterogêneas.

3.2 Descritores de Textura

No contexto de CBIR a textura refere-se a um padrão de repetição de um determinado elemento ou vários elementos em diferentes posições relativas. Geralmente, a repetição envolve variações locais de escala, orientação, ou outras características geométricas e ópticas dos elementos.

[22] descreve as medida de textura como outro importante fator na diferenciação de objetos pela visão humana e por esta razão tem sido incorporada com outra característica dos descritores no intuito de aprimorar a forma de representação das imagens. Estes descritores são comumente utilizados nos domínios de aplicação onde as imagens são ricas em detalhes e geralmente com pouca informação de cor. Esta categoria de descritores utilizam a propriedade de uma determinada região que descreve o padrão de variação de tons de cinza e cor numa determinada área, como visto em [2]. É importante observar que, devido a grande quantidade de operações necessárias para cada pixel da imagem, o custo computacional destes descritores é bem mais elevado que quando comparado aos descritores de cor e forma.

[6] propuseram dois métodos o primeiro chamado Color and Edge Directivity Descriptor - CEDD e o segundo Fuzzy Color and Texture Histogram - FCTH. Segundo [26] foram comprovados ganhos substanciais destes dois descritores sobre os descritores propostos no padrão MPEG-7. O primeiro descritor CEDD, inclui a informação de textura produzida pelo histograma de seis bins do sistema fuzzy que utiliza os cinco filtros digitais proposto no padrão MPEG-7. Além disso, para informação de cor do CEDD utiliza o histograma de cores de 24-bin produzida pelo sistema fuzzy-linking de 24-bin. Desta forma o histograma final tem $6 \times 24 = 144$ regiões. Cada bloco de imagem sucessivamente interage com todos os sistemas fuzzy. Definindo o bin produzido pelo sistema fuzzy de informação de textura como n e o bin produzido pelo sistema fuzzy-linking de 24-bin de m , em seguida cada bloco de imagem é colocada na posição bin: $n \times 24 + m$. Como medida de similaridade é utilizado o coeficiente de Tanimoto. O descritor FCTH inclui a informação de textura produzida no histograma de 8-bins do sistema fuzzy que utiliza as bandas de alta frequência da transformada Haar Wavelet. Para obter informações sobre as cores, o descritor utiliza o histograma de cores de 24-bin produzido pelo sistema de 24-bin fuzzy-linking. De modo geral, o histograma final inclui $8 \times 24 = 192$ regiões. Para a realização dos experimentos foram usadas as seguintes bases de imagens: Wang (1000

imagens), MPEG-7 CCD (aproximadamente 5000 imagens), UCID (1300 imagens) e NISTER (10200 imagens). Observa-se que estas bases de imagens não representam a heterogeneidade e tão pouco possuem um tamanho significativo.

[28] é proposto um algoritmos simples com bons resultados de eficácia e invariante à rotação e variações na escala de cinza. É definido com uma janela com raio R e uma quantidade de vizinhos P e percorre a imagem calculando a quantidade de variações positivas ou negativas entre os brilhos dos pixels vizinhos em relação ao pixel central da janela. Somente o sinal da variação é registrado: 1 quando a variação é positiva e 0 quando a variação é negativa isso garante a invariância a variações na escala de cinza. Também é feita uma contagem de quantas transições entre 0/1 e 1/0 existem na vizinhança garantindo invariância a rotação. Se a quantidade de transições for menor ou igual a 2, o valor de LBP para aquela posição da janela é igual a quantidade de sinais 1 na vizinhança, caso contrário, o valor de LBP é $P + 1$, ao final do percurso da janela pela imagem, cada posição possui um valor LBP entre 0 e $P + 1$. Calcula-se um histograma dos valores LBP contendo $P+2$ bins.

3.3 Descritores de Forma

De acordo com [41], esta classe de descritores é dividida em: Baseado em contorno e Baseado em regiões. Cada uma dessas sub-classes é dividida em estrutural ou global. Note que cada uma destas classes de descritores de forma é aplicado a um domínio específico de aplicação como: Determinar o contexto da forma do objeto, categorização de caracteres, análise de imagens médicas, detecção de borda de imagens, encontrar um determinado ponto em uma imagem, determinar o centro de gravidade de um objeto na imagem ou da imagem, dentre vários outros. Citaremos alguns dos descritores de forma existentes na literatura.

[11], propôs os métodos Contour Saliences (CS) e Segment Saliences (SS). O CS utiliza as saliências da forma do objeto para representá-lo, estas saliências de forma

são definidas como pontos de maior curvatura ao longo do contorno do objeto, este é utilizado como referência para o cálculo dos demais pontos completando assim a representação do vetor de características. O método SS é uma variação do método anterior, porém nesta proposta não considera-se apenas pontos de contorno e sim em um número pré-definido de segmentos de contorno com o mesmo tamanho. As áreas de influência internas e externas de cada segmento são computadas somando-se as áreas de influência de seus pixels correspondentes. Um segmento é considerado convexo se sua área acumulada externa é maior do que sua área acumulada interna e, no caso contrário, é considerado côncavo.

O método Curvature Scale Space, proposto por [1], é outro método amplamente utilizado nos domínios de aplicação onde a descrição da forma da imagem é necessária. Este é um descritor que obtém os contornos de um determinado objeto na imagem através de sucessivas suavizações por uma função Gaussiana e onde cada estágio desta suavização do contorno representa uma escala na curva.

O método *Edge Histogram Descriptor - EHD* proposto por [32] representa a distribuição de 5 tipos de borda, ou seja vertical, horizontal, 45° diagonal, de 135° diagonal, e sem borda. Para gerar a representação EHD, uma imagem é dividida em 4×4 sub-imagens não sobrepostas. Então cada sub-imagem serve como uma região base para gerar o histograma de borda, que consiste de 5 bins com os tipos vertical, horizontal, 45° , 135° e sem borda. Uma vez que existam 16 (4×4) sub-imagens, cada imagem gera um histograma de borda com um total de 80 bins. Estes 80 bins quantizados e normalizados constituem o padrão EHD do MPEG.

Devido as características e aplicabilidade específicas dos descritores de forma e devido a grande heterogeneidade das imagens dispostas na Web, o uso exclusivo de descritores de forma para CBIR em bases grandes e heterogêneas é desaconselhado.

3.4 Bag of Visual Words/Features

Bag of visual words/features - (*BoVW*) não deve ser visto como uma classe de descritores, e sim como uma metodologia que utiliza os descritores de imagem (descritores de cor, forma, textura e/ou a combinação destes) no seu processo de representação da imagem. É importante ressaltar que esta abordagem é usualmente empregada para as tarefas de **classificação de imagens** e **detecção de objetos na imagem**. Porém, atualmente existem algumas iniciativas para se recuperar imagens similares à uma imagem dada como consulta utilizando os princípios gerais das Bag of Visual Word/Features.

A idéia básica desta metodologia é representar uma imagem como um documento visual composto de elementos visuais distintos, similar a idéia de palavras em textos. Basicamente esta metodologia consiste em detectar propriedades da imagem, criar um dicionário de termos (de acordo com as propriedades detectadas) e representá-las pelo agrupamento dos termos do dicionário presentes na imagem. A criação do dicionário é feita representando os pontos de interesse/patches de acordo, geralmente, com um descritor baseado no Scale-invariant feature transform - SIFT proposto por [21]. Na abordagem *BoVW*, a comparação entre duas imagens é feita utilizando a distância Euclidiana.

O atual estado da arte em *bag of features* é a abordagem proposta por [15] em seu artigo Improving Bag-of-Features for Large Scale Image Search, que consistem em melhorar o padrão de representação das bag-of-features utilizando duas modificações significativas. A primeira é baseada no encapsulamento de Hamming, que fornece assinaturas binárias para refinar as palavras visuais. Resultando em uma medida de similaridade para os descritores atribuídos à mesma palavra visual. A segunda é um método que impõe restrições de consistência geométrica e utiliza conhecimento sobre as transformações de rotação e escala das imagens. As restrições são integrados no arquivo invertido e são aplicados a todas as imagens do banco de dados. Foi observado que em ambos os métodos existe uma melhora significativa.

tiva na performance, especialmente para grandes base de dados sem um aumento do tempo de execução. Para os experimentos foram usadas três diferentes bases classificadas, uma base criada pelos autores e citada em uma publicação anterior dos próprios autores ([14]) chamada de Holidays dataset, a Oxford5k e University of Kentucky object recognition benchmark inicialmente citada por [25].

O método *Speeded Up Robust Features - SURF* proposto por [3] é um conhecido detector e descritor de pontos de interesse na imagem. Segundo [10] o SURF é amplamente utilizado em aplicações de visão computacional. Por se tratar de um detector/descritor de pontos de interesse confirmamos que para busca de imagens em bases grandes e heterogêneas este método puro não obtém bons resultados, conforme discutido no Capítulo 5.3.

3.5 Modelo Vetorial

O modelo de espaço vetorial [29], ou simplesmente modelo vetorial, representa documentos e consultas como vetores de termos. Termos são ocorrências únicas nos documentos. Os documentos devolvidos como resultado para uma consulta são representados similarmente, ou seja, o vetor resultado para uma consulta é montado através de um cálculo de similaridade.

Aos termos das consultas e documentos são atribuídos pesos que especificam o tamanho e a direção de seu vetor de representação. Ao ângulo formado por estes vetores dá-se o nome de θ . O $\cos \theta$ determina a proximidade da ocorrência. O cálculo da similaridade é baseado neste ângulo entre os vetores que representam o documento e a consulta, através da seguinte equação:

$$\text{sim}(d, q) = \frac{\sum_{i=1}^t w_{id} \times w_{iq}}{\sqrt{\sum_{i=1}^t w_{id}^2} \times \sqrt{\sum_{i=1}^t w_{iq}^2}}$$

Os pesos quantificam a relevância de cada termo para as consultas (w_{iq}) e para os documentos (w_{id}) no espaço vetorial. Para o cálculo dos pesos w_{iq} e w_{id} , utiliza-se uma técnica que faz o balanceamento entre as características do documento,

utilizando o conceito de frequênciade um termo num documento. Se uma coleção possui N documentos e n_{t_i} é a quantidade de documentos que possuem o termo t_i , então o inverso da frequênciade termo na coleção, ou idf (inverse document frequency) é dado por:

$$idf_i = \log \frac{N}{n_i}$$

Este valor é usado para calcular o peso do termo, utilizando a seguinte equação:
 $W_{id} = freq(t_i, d) \times idf_i$ ou seja, é o produto da frequênciade termo no documento pelo inverso da frequênciade termo na coleção.

Capítulo 4

SDLF - Sorted Dominant Local Feature

O descriptor proposto neste trabalho inclui um processo com 3 processos bem definidos:

Passo I Extração das propriedades das imagens da base e das imagens de consulta;

Passo II Codificação das propriedades extraídas das imagens para a representação textual;

Passo III Utilização do modelo vetorial [29] para indexação e busca de imagens, em sua representação textual.

Dado que a etapa de codificação das propriedades extraídas das imagens estar intrinsecamente relacionada à etapa de extração das propriedades das imagens da base e das imagens de consulta, ao explicar como a representação textual é gerada, estaremos explicando as duas primeiras etapas concomitantemente. Pelo efeito meramente didático, iremos explicar o método proposto utilizando apenas as propriedades de cor. No intuito de manter o fluxo da leitura e entendimento contínuo, a intuição por trás das decisões de cada passo serão explicados na Seção 4.1.

Passo I

Seja \hat{I} uma **imagem**, definida como um par (D_I, \vec{I}) , onde:

- D_I é um conjunto finito de *pixels* (pontos em \mathbb{N}^2 , isto é, $D_I \subset \mathbb{N}^2$), e
- $\vec{I}: D_I \rightarrow \mathbf{D}'$ é uma função que atribui cada pixel p em D_I a um vetor $\vec{I}(p)$ de valores em algum espaço arbitrário \mathbf{D}' (por exemplo, $\mathbf{D}' = \mathbb{R}^3$ onde a cor no sistema RGB é atribuída a cada pixel).

Seja $\mathcal{R} = \{r_1, r_2, \dots, r_\eta\}$ um conjunto de η regiões de \hat{I} , tal que cada pixel p em D_I pertence a apenas uma região $r_j \in \mathcal{R}$, como mostrado na Figura 4.1(a).

Passo II

Seja $\mathcal{B}_{r_j} = \{b_1.r_j, b_2.r_j, \dots, b_\beta.r_j\}$ uma partição da região r_j consistindo de blocos de pixels consecutivos não sobrepostos, Figura 4.1(b).

Para cada bloco $b_i.r_j \in \mathcal{B}_{r_j}$, nós representamos seu conteúdo como um *código textual*

$$t_{i,j} = \langle \hat{c}_1 - \hat{c}_2 - \dots - \hat{c}_n \rangle (n \geq 0),$$

onde \hat{c}_k é a identificação da cor tal que $\mathcal{H}_{i,j}(\hat{c}_k) \geq \epsilon$ e $(\hat{c}_k) > (\hat{c}_{k+1})$, para $1 < k < n-1$, sendo ϵ um limiar constante positivo, e $\mathcal{H}_{i,j}$ sendo histograma de cor do bloco $b_i.r_j$ (Figura 4.1(c)), mapeando cada identificação de cor à sua frequência no bloco, como mostrado na Figura 4.1(d).

Passo III

A terceira etapa do método consiste em implementar o modelo vetorial para indexação e busca das imagens. No modelo vetorial, o vocabulário da coleção desempenha o mesmo papel que o codebook [16, 18] nos métodos BoVW [20, 35], este define a dimensão do espaço onde as imagens são representadas.

Dado um vocabulário com V distintos códigos textuais, cada imagem i da coleção é representada como um vetor $\hat{v}_i = (w_{i,1}, w_{i,2}, \dots, w_{i,V})$. Cada dimensão de \hat{v}_i corresponde ao peso de um código textual distinto do vocabulário na imagem i . O peso do código textual j em i ($w_{i,j}$) é definido como:

$$w_{i,j} = tf(t_{i,j}) \times idf(t_{i,j}) \quad (4.1)$$

onde $tf(t_{i,j})$ é a frequência do código textual $t_{i,j}$ na imagem, e $idf(t_{i,j})$ é a medida da importância de uma ocorrência de um código textual na coleção, sendo computado como:

$$idf(t_{i,j}) = \log\left(\frac{N}{df(t_{i,j})}\right), \quad (4.2)$$

onde N é o número de imagens na coleção e $df(t_{i,j})$ é o número de imagens em que $t_{i,j}$ aparece.

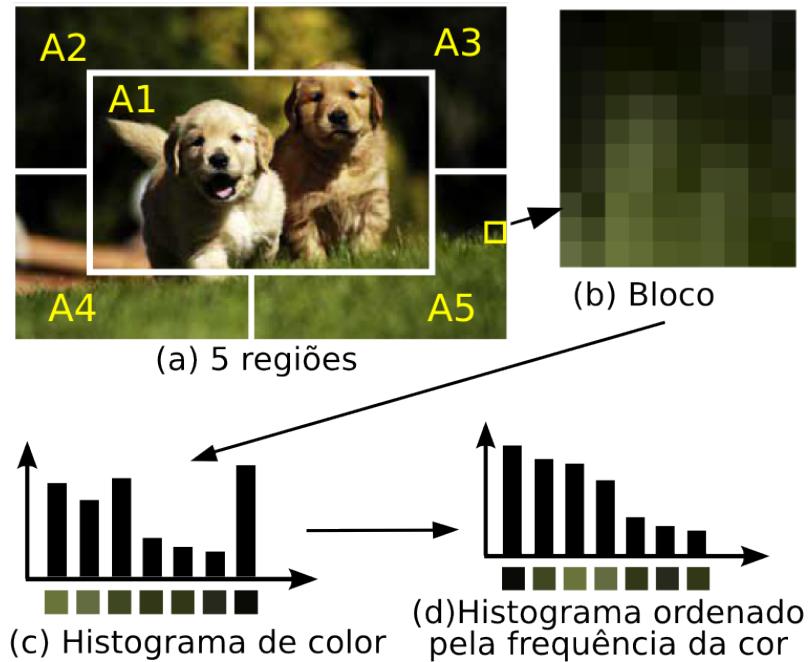


Figura 4.1: *Exemplo de extração do código textual.*

Para computar a similaridade entre duas imagens A e B , nós usamos a distância do coseno como medida [29], definida como segue:

$$\cos(A, B) = \frac{\sum_{i=1}^n \hat{v}_{A_i} \times \hat{v}_{B_i}}{\sqrt{\sum_{i=1}^n (\hat{v}_{A_i})^2} \times \sqrt{\sum_{i=1}^n (\hat{v}_{B_i})^2}} \quad (4.3)$$

4.1 Detalhamento dos passos do método proposto

Como mencionado na seção anterior, aqui detalharemos as intuições por trás em cada etapa do método. No **Passo I** optamos por dividir a imagem em 5 grandes regiões, como mostra a Figura 4.1(a). Optamos por este formato para que ficasse clara a distinção entre os códigos textuais gerados em cada região da imagem. Observe os dois blocos exibidos na Figura 4.2, por exemplo. Este é o típico caso onde observamos que os códigos textuais gerados para região superior da imagem serão os mesmos para os blocos da região inferior, dado que as cores mais representativas de cada bloco serão as mesmas após o processo de redução da quantização. Em ambos os casos observamos que o código textual gerado é '6x16x53x59x132', por este motivo para diferenciá-los usamos os conceito proposto por [17] onde são usadas 5 regiões fixas como mostra a figura 4.2. Desta forma os blocos passam a ter a seguinte representação textual respectivamente, 'A2x6x16x53x59x132' e 'A4x6x16x53x59x132'. Um outro exemplo é o das Figuras 4.3 e 4.4 onde dois blocos com a representação textual idêntica estão em regiões diferentes.

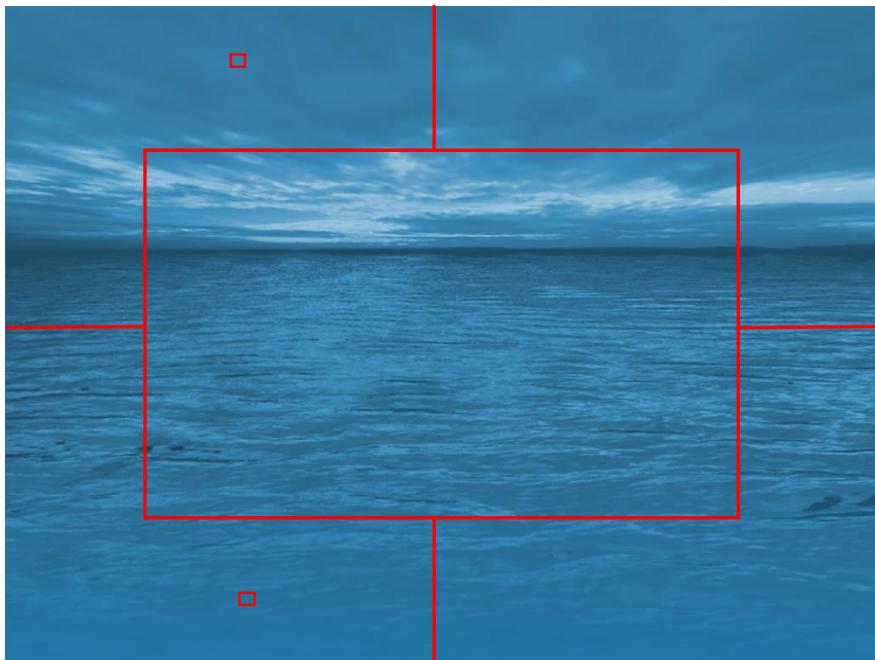


Figura 4.2: *Exemplo de extração do código textual.*

O **Passo II** é onde efetivamente os códigos textuais são gerados e para isso,

como visto no Capítulo anterior, I. extraímos o histograma de cores do bloco, II. ordenamos o histograma baseado na frequência de cores e III. eliminamos as cores com menor frequencia. A razão pela qual ordenamos o histograma de cada bloco reside no fato de que as cores mais frequentes em cada bloco, são naturalmente as cores mais representativas do bloco.

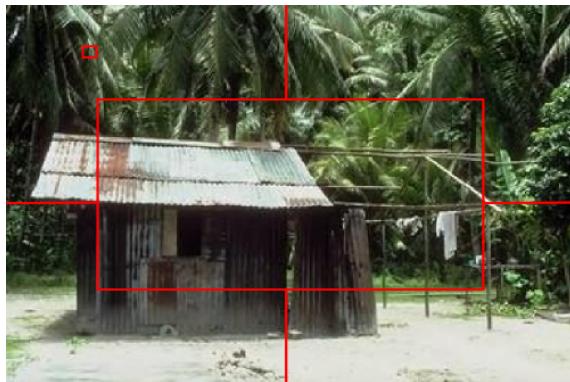


Figura 4.3: Bloco na região A2



Figura 4.4: Bloco na região A5

O limiar ϵ , utilizado para eliminar as cores de menor frequência, é usado para que tais cores não gerem ruído durante o processo de geração dos códigos textuais, o que reduz o tamanho dos códigos gerados e aproxima as representações textuais. Por exemplo considere os seguintes códigos textuais *a*. A1x123x34x122x28x45x15x10 e *b*. A1x123x34x122x28x45, estes códigos são claramente códigos distintos, no entanto, considere que as cores 15 e 10 ocorrem com uma frequência menor que o limiar e desta forma serão eliminadas, assim os códigos textuais *a* e *b* seriam respectivamente A1x123x34x122x28x45 e A1x123x34x122x28x45. O limiar ϵ , assim como o número de blocos usados em nossos experimentos foram adotados empiricamente e serão explicados no Capítulo 5.

Capítulo 5

Experimentos

Neste capítulo detalharemos os experimentos realizados para a validação do método proposto neste trabalho. Para a realização de nossos experimentos utilizamos quatro bases de imagens: (1) *WANG*, proposta por [37], (2) *MPEG-7 Common Colour Dataset - CCD*, (3) uma base de imagens coletadas de sites de comércio eletrônico, de agora em diante chamada de *ROUPAS*, e (4) uma base de aproximadamente 104.000 imagens genéricas coletadas do website Yahoo!, que denominamos 100k.

Como baseline foram utilizados os seguintes descritores de imagem baseados em cor: *Color and Edge Directivity Descriptor - CEDD* e *Fuzzy Color and Texture Histogram - FCTH* ambos apresentados por [7], os descritores do padrão MPEG-7 todos apresentados por [32], *Scalable Color Descriptor - SCD*, *Edge Histogram Descriptor - EHD* e *Color Layout Descriptor - CLD* e o descritor baseado em pontos de interesse o *Speeded Up Robust Features - SURF* proposto por [3]. Os descritores CEDD e FCTH foram escolhidos por possuírem os melhores resultados quanto eficiência e eficácia dentre os métodos baseados em propriedades de cor da imagem, conforme visto em [7] e [26]. Os descritores do padrão MPEG-7 foram escolhidos por serem amplamente citados na literatura. O método *SURF* foi adotado com a intenção de ratificar a ideia de que os métodos padrão de detecção de pontos de interesse, quando usados no contexto de CBIR em bases de imagens genéricas, não

são indicados, como mostram os resultados dos experimentos.

No intuito de comprovar a eficiência e eficácia do *Sorted Dominant Local Feature -SDLF* em um domínio diferente do domínio de cor, utilizamos o descritor de textura *Local Binary Pattern - LBP* como descritor de características para o processo de geração dos códigos textuais como visto na Figura 4.1(c) do Capítulo 4.

Como métrica de comparação utilizamos a medida *MAP - Mean Average Precision*, dado que queremos que as respostas mais relevantes estejam no topo do rank de respostas.

5.1 Bases de imagens

Como mencionado anteriormente, neste trabalho utilizamos as bases *Wang* e *CCD* por serem amplamente citadas no contexto acadêmico. Usamos também a base *ROUPAS* por caracterizar bem o domínio de textura, e usamos a base *100k* como a fonte de ruído no intuito de reproduzir o ambiente Web. Abaixo segue uma breve explicação sobre cada uma delas.

A base *WANG* é constituída por 1000 imagens coloridas agrupadas em 10 conjuntos de 100 imagens similares. Pelo fato da base de imagem *WANG* não possuir um conjunto explícito de imagens de consulta utilizamos a seguinte abordagem: Para cada um dos 10 conjuntos de 100 imagens escolhemos aleatoriamente 5 imagens de consulta (Figura 5.1), e consideramos como conjunto de resposta todas as 100 imagens de cada um dos conjuntos pré-definidos.

A base *Common Color Dataset - CCD* é constituída por 5.466 imagens coloridas e possui um conjunto com 50 imagens de consulta (Figura 5.2) previamente definidas com seus respectivos conjuntos de imagens similares, deste ponto em diante chamada de *ground truth*.

A base *100k* é Composta por 103.348 imagens coloridas com diferentes resoluções e temas, todas coletadas do diretório Yahoo!, desta forma montamos uma base



Figura 5.1: Consultas utilizadas para base WANG.

de imagens heterogênea e de maior tamanho que as demais. A heterogeneidade e tamanho desta base simula um processo de busca na Web, onde há um número elevado de imagens e não há um padrão bem determinado nas propriedades destas imagens.

Com o intuito de comprovar a eficiência e eficácia do nosso método em um ambiente de consulta real utilizamos a seguinte abordagem para a configuração das bases usadas nos experimentos: Inicialmente dividimos aleatoriamente a base 100k em grupos de 10.000, 20.000, 30.000, ..., 100.000 imagens (de agora em diante

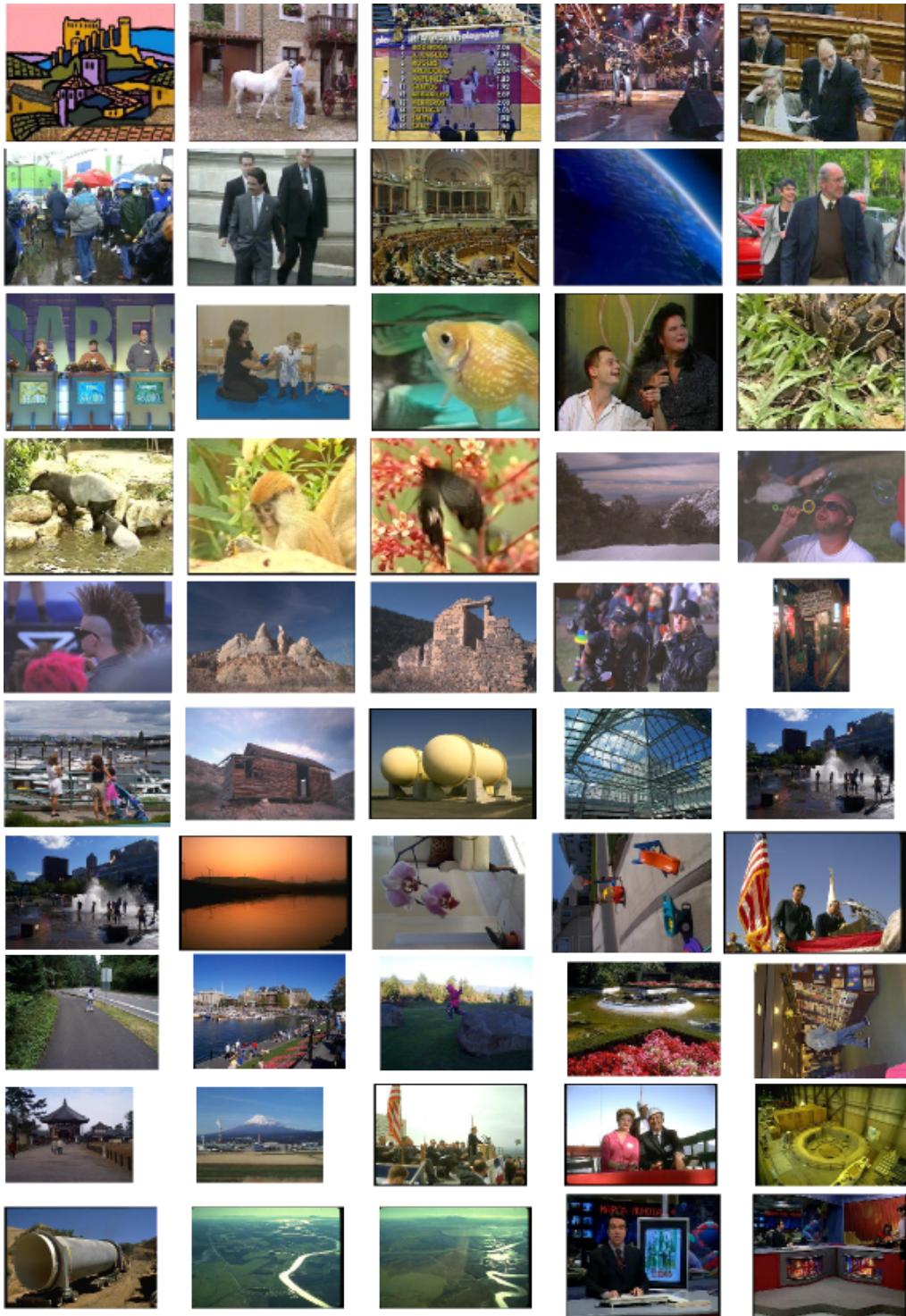


Figura 5.2: Consultas utilizadas para base CCD.

chamadas de 10k, 20k, 30k, etc), então inserimos em cada um dos grupos a base *WANG* inteira. Repetimos este processo para a base CCD. Como resultado final teremos 20 bases. Sendo 10 bases referentes à combinação da base *WANG* com a

100k nos tamanhos de 11.000, 21.000, ..., 101.000 imagens e 10 bases referentes à combinação da base *CCD* com a base 100k nos tamanhos de 15.000, 25.000, ..., 105.000 imagens.

A base *Roupas* é constituída por 5.000 imagens divididas em 5 categorias: quadriculado, estampa, lisa, lisa com botão e listras. Foram escolhidas aleatoriamente 50 imagens de consulta, 10 imagens de cada categoria como mostra a (Figura 5.5). Esta base foi criada com a intenção de validar o método em um domínio de aplicação específico que utilize propriedades de textura.

5.2 Experimentos de configuração do SSDLF

Conforme mencionado no Capítulo 4, encontramos empíricamente os valores para o número de blocos e para o limiar ϵ . Executamos estes experimentos sobre 10% de imagens da base aleatoriamente selecionadas.

O número de blocos a ser utilizado pelo *SSDLF* é obtido dividido-se a imagem de forma que o número de blocos na horizontal seja igual ao número de blocos na vertical. As dimensões dos blocos variam de acordo com as dimensões de cada imagem, um exemplo de bloco pode ser visto na Figura 4.1(b) do Capítulo 4.

Para se encontrar o número ideal de blocos utilizamos as seguintes configurações em nossos experimentos preliminares: 27×27 (729 blocos), 32×32 (1024 blocos), 36×36 (1296 blocos), 45×45 (2025 blocos) e 50×50 (2500 blocos por imagem). Para cada uma destas configurações utilizamos os seguintes limiares ϵ 2%, 5% e 10% das cores menos frequentes, conforme mostra a Figura 5.3.

É importante observar que independente das dimensões da imagem (Altura x Largura), o número de blocos se mantém constante se adaptando à forma da imagem conforme mostra a Figura 5.4.

Observamos um aumento significativo na qualidade das respostas até a configuração de 1296 blocos com limiar em 5%, a partir deste ponto houve uma estabi-

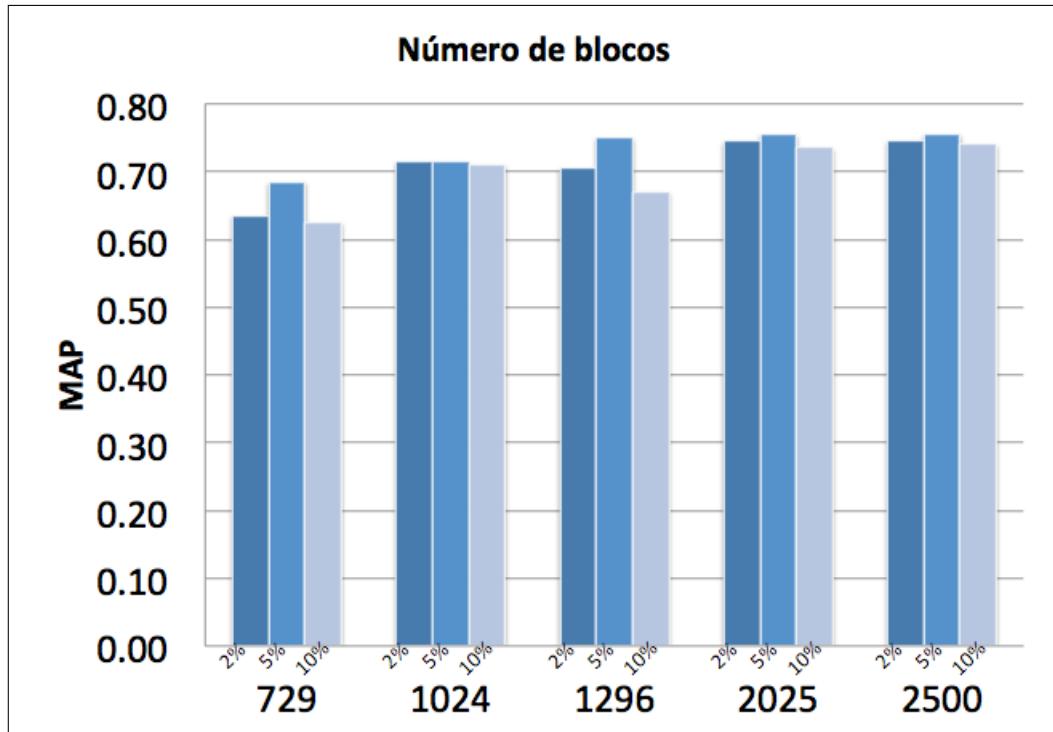


Figura 5.3: Escolha do número de blocos e limiar.

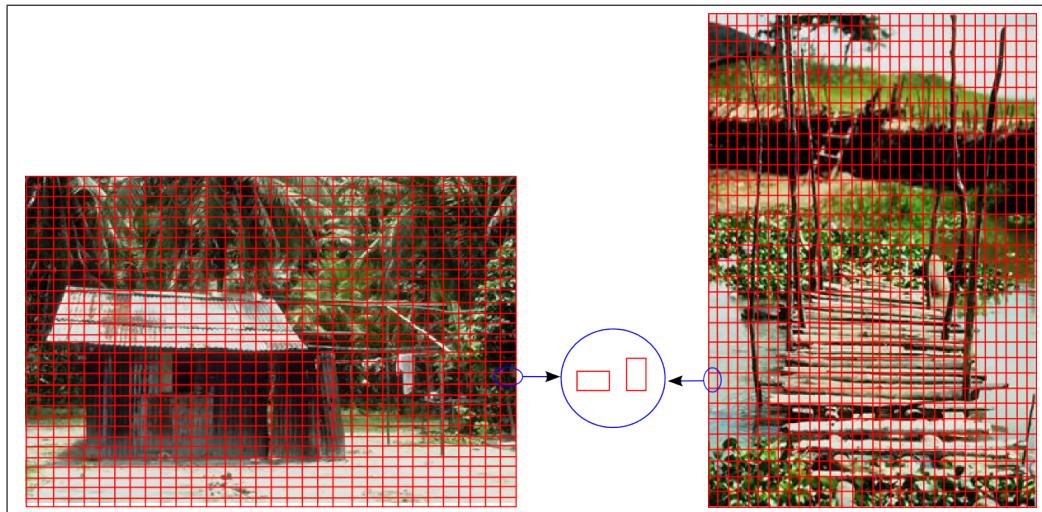


Figura 5.4: Blocos se adaptam às dimensões da imagem.

lização na qualidade das respostas. No entanto o maior número de blocos implica no aumento do custo computacional, por esse motivo adotamos 1296 blocos por imagem.

Após a escolha do número de blocos e limiar adequado, para os experimentos em questão, iniciamos a etapa de realização dos experimentos descrita a seguir.



Figura 5.5: Consultas utilizadas para base ROUPAS.

5.3 Experimentos

Nesta seção iremos comparar a eficácia do método proposto com os métodos existentes na literatura baseados em cor para CBIR.

Dado que as bases *WANG* e *CCD* possuem características distintas, tivemos que conduzir os experimentos da seguinte forma:

Base *WANG*, como visto anteriormente esta base tem 10 grupos de 100 imagens rotuladas incrementalmente de 0.jpg a 999.jpg e agrupadas de 100 em 100

imagens. Para cada grupo 10k, 20k, 30k, ..., 100k foram realizados os experimentos com as mesmas imagens de consulta e os resultados foram obtidos calculando a média das médias dos resultados de cada grupo.

Base *CCD*, esta base possui um *ground truth* previamente definido por especialistas e 50 imagens de consulta. Os resultados obtidos pelos métodos testados com cada uma das imagens de consulta foram comparados com o *ground truth* desta base e calculamos a média de suas respostas para as 50 consultas em cada um dos 10 grupos (10k, 20k, 30k, ..., 100k). Em ambas as bases utilizamos como descritor de imagem as propriedades de cor como explicado no Capítulo 4.

Os resultados obtidos pelos métodos testados com cada uma das imagens de consulta foram comparados com o *ground truth* desta base e calculamos a média de suas respostas para as 50 consultas.

Base *ROUPAS*, esta base possui um *ground truth* previamente definido por especialistas e 50 imagens de consulta. Note que nestes experimentos queremos comprovar que o *SDLF* pode ser configurado com um descritor de imagem baseado em outras propriedades, neste caso, textura. Os experimentos aqui foram realizados como segue:

- Executamos o $LPB_{8,1}$ sobre a base inteira e sobre as imagens de consulta;
- Utilizamos a distância Euclidiana para comparar os vetores das imagens de consulta com os vetores da base, rankear as respostas e pegar as top 10 respostas;
- O *SDLF* foi configurado substituindo o histograma de cor ($\mathcal{H}_{i,j}$) do bloco por seu $LPB_{8,1}$, deste ponto em diante o método continua inalterado;
- Executamos o *SDLF* modificado sobre a base de imagens e sobre as imagens de consulta;
- Utilizamos o modelo vetorial para indexar, buscar, rankear as respostas e pegamos as top 10 respostas;
- Como métrica de comparação utilizamos o MAP.

Como controle também executamos o *CEDD* e *FCTH* sobre esta base, a Figura 5.6 mostra os resultados destes experimentos.

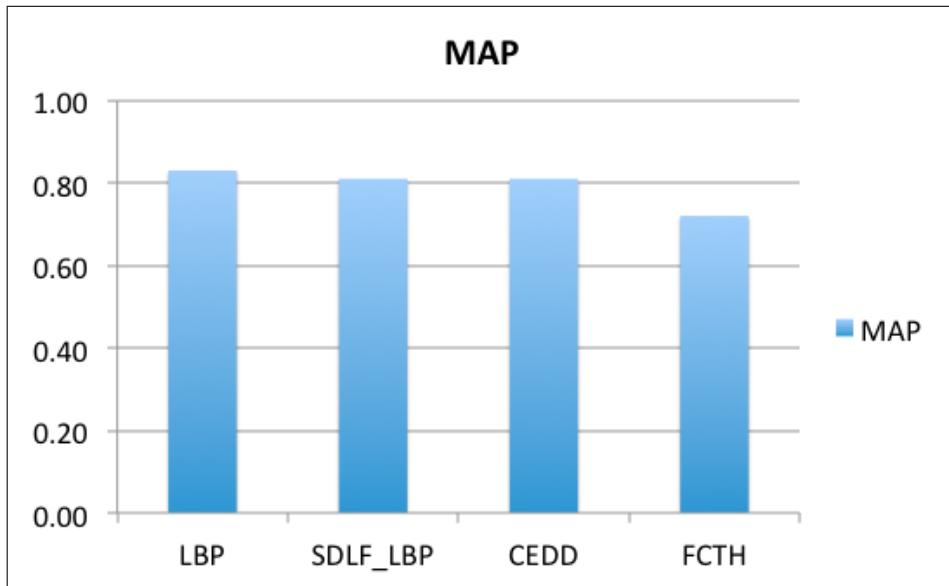


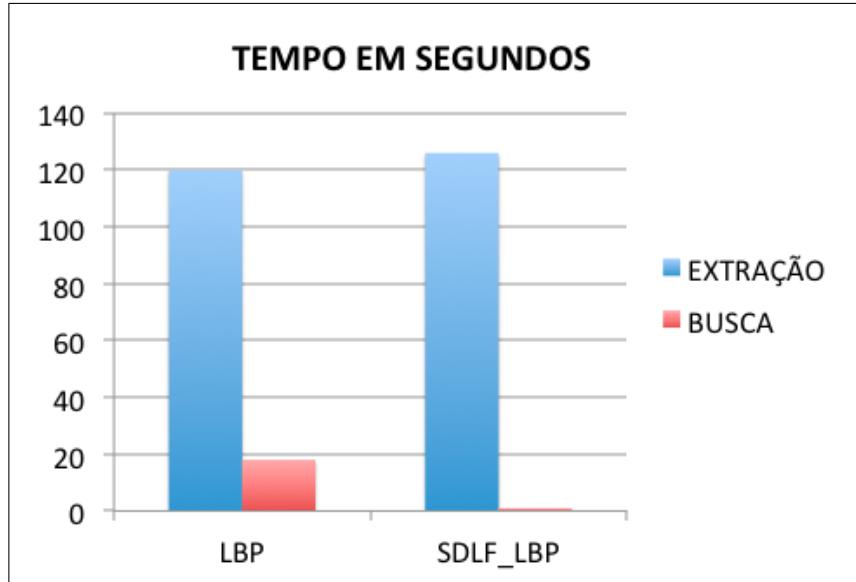
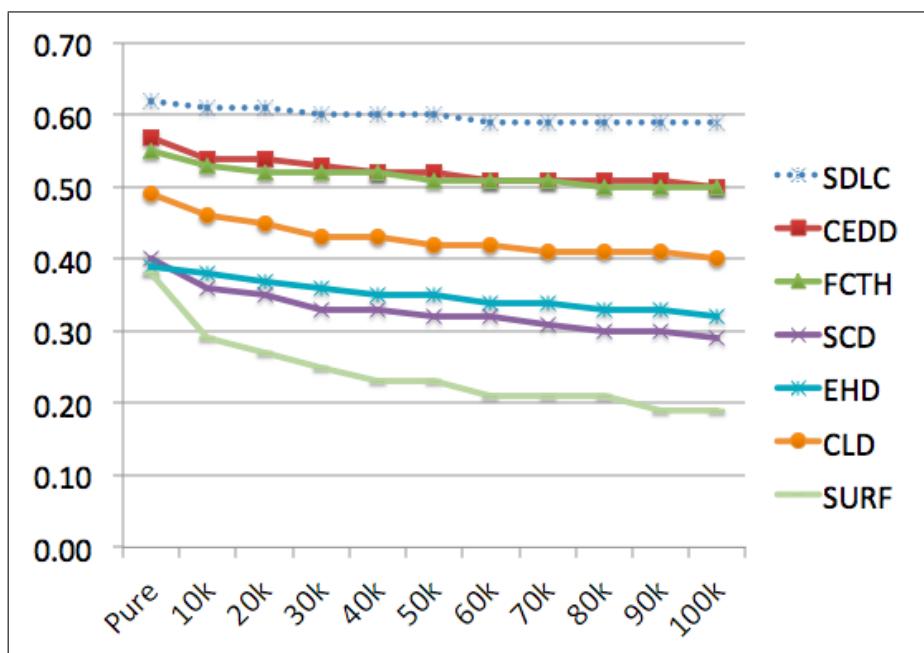
Figura 5.6: Resultados obtidos com a base *ROUPAS*

Verificamos que não houve diferença estatística significativa entre os resultados do *SDLF* modificado e os outros métodos. Dado que os métodos *CEDD* e *FCTH* foram os que obtiveram os melhores resultados dentre os métodos utilizados como *baseline*, aqui a comparação foi feita apenas com esses dois métodos. Observamos também um expressivo ganho quanto ao tempo de extração e busca, utilizando o *SDLF* modificado sobre o *LBP* puro, como apresentado na Figura 5.7.

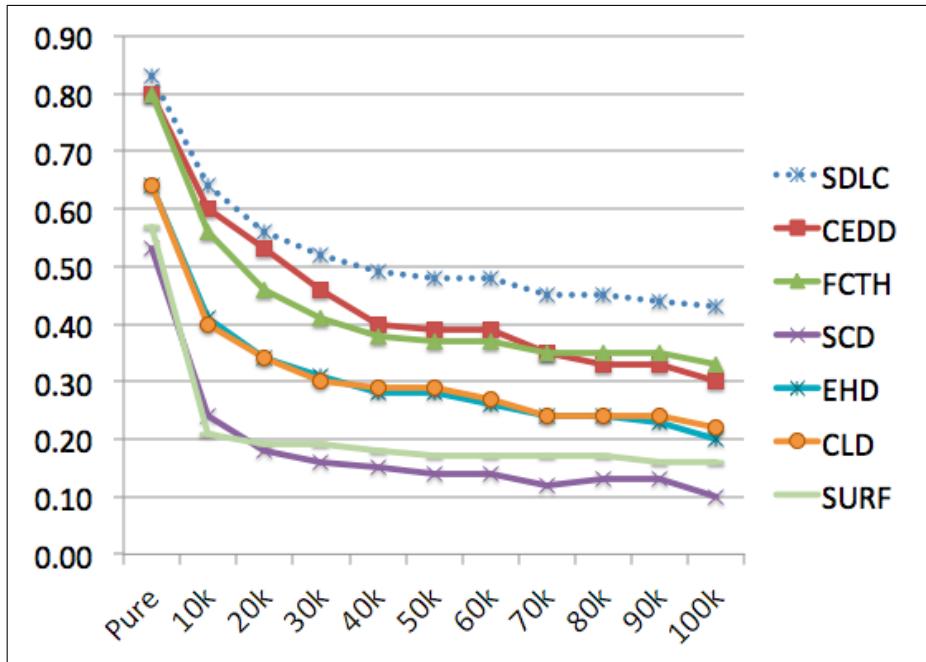
Nossos experimentos foram validados estatisticamente através dos testes propostos por [38]. Tal validação foi usada para comprovar que a diferença entre os resultados obtidos por todos os métodos experimentados são estatisticamente significativos.

Nas Figuras 5.9 e 5.8 pode-se observar que o método *SDLF* é superior aos demais métodos implementados. Sendo esta superioridade comprovada estatisticamente a partir da base 30k, para ambos os casos *WANG* e *CCD*, onde os resultados obtiveram significância estatística $\geq 98\%$.

Estes resultados comprovam que o descritor proposto neste trabalho é mais ro-

Figura 5.7: Resultados obtidos com a base *ROUPAS*Figura 5.8: Resultados obtidos com a base *CCD*

busto quanto a quantidade, variação no tamanho, qualidade e diversidade das imagens da base. Dados que os resultados obtidos pelo descritor proposto foram superiores aos métodos amplamente discutidos na literatura, concluímos que este pode ser usado como base para os trabalhos futuros de criação de um descritor de imagens adaptável a diferentes tarefas de busca.

Figura 5.9: Resultados obtidos com a base *WANG*

5.4 Discussão

É importante observar que utilizando cores como as únicas propriedade descritoras da imagem existe a necessidade de um processo de quantização de cores eficiente. No entanto, a despeito do processo de quantização utilizado, em muitos casos imagens que deveriam ser retornadas pelo processo de consulta são descartadas, em alguns casos, devido a uma leve variação de cor, brilho ou saturação. Observe o exemplo da Figura 5.10, onde a imagem da esquerda é a imagem de consulta. A imagem da direita está presente na base, porém não é retornada pelo método.



Figura 5.10: Falha no método para o caso do descritor baseado exclusivamente nas propriedades de cor das imagens.

Capítulo 6

Conclusões

Como continuação do trabalho proposto temos em vista os seguintes passos:

- Concluir e submeter de uma artigo para a *International Conference on Image Processing - ICIP'12* mostrando os resultados obtidos nos experimentos realizados com o descritor proposto SDLC;
- Expandir e submeter o artigo submetido ao ICIP para o *Journal of the American Society for Information Science and Technology - JASIST*;
- Realizar o estudo das propriedades das bases de imagem que fornecem informações que possibilitam o melhor ajuste do descritor proposto para o processo de CBIR. Entendemos que o melhor ajuste no descritor resulta em um aumento da sua eficácia na busca de imagens em bases de domínio específico. Para isso propomos a hipótese da heterogenidade da base.
- Trabalhar o conceito de Heterogeneidade, onde sugerimos que quanto maior o grau de heterogeneidade da base mais relacionada a aplicações genéricas ela é, como por exemplo a Web. Por outro lado, quanto menor o grau de heterogeneidade da base mais "comportada" é a base, por exemplo possuindo categorias de imagens bem definidas ou sendo bases de domínio específico.

Conforme discutido no Capítulo ??, para o cálculo do grau de heterogeneidade, serão usadas as seguinte propriedades da imagem: espaço de cor, tamanho em pixels, propriedades de cor, forma e textura e a densidade da base de imagem. Este último conceito é baseado na idéia de que podemos agrupar as imagens da amostra com algum algoritmo de agrupamento, por exemplo o k-means, utilizando como distância entre as imagens os resultados dos melhores descritores de cor, forma e textura. Conforme o número de agrupamentos gerados para cada descritor poderemos afirmar que a base de imagens em questão possui propriedades de cor, forma ou textura, informação fundamental para a escolha e/ou ajuste do descritor a ser utilizado no processo de CBIR.

- Implementar de um descritor que permita ajustes conforme as características da base. Tais ajustes podem ser feitos automaticamente, uma vez que as propriedades da base sejam detectadas.
- Realizar experimentos nos espaços de cor HSI (*hue, saturation, intensity*) e HSV (*hue, saturation, value*) para verificar o comportamento do método proposto nesses espaços de cor. Acreditamos que no espaço de cor HSV os resultados dos experimentos sofrerão uma relativa melhora.
- Realizar novos experimentos utilizando novos métodos de quantização. Os experimentos realizados até o momento foram conduzidos utilizando a quantização no padrão Web com 216 cores. Nos próximos experimentos iremos utilizar o padrão de quantização linear de 256 cores nos dois espaços de cor HSV e CIELAB.
- Realizar experimentos em bases maiores como *MIRFlicker* proposta por [13] composta por 1.000.000 de imagens, imageCLEF¹ composta por 237.000 imagens coletadas da Wikipedia e Nhemu composta por imagens de comércio eletrônico.

¹<http://www.imageclef.org/2011/wikipedia>

- Realizar todos os experimentos com tomada de medida de tempo de extração, tempo de consulta e tamanho do vetor de características.

Bibliografia

- [1] S. Abbasi, F. Mokhtarian, and J. Kittler. Enhancing css-based shape retrieval for objects with shallow concavities. *Image and Vision Computing*, 18(3):199 – 211, 2000.
- [2] A. Baraldi and F. Parmigiani. An investigation on the texture characteristics associated with gray level co-occurrence matrix statistical parameters. *IEEE Transaction on Geosciences and Remote Sensing*, 32(2):293–303, 1995.
- [3] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *In ECCV*, pages 404–417, 2006.
- [4] N. Chang and K. Fu. Query-by-pictorial-example. *TSE, SE-6(6)*:519–524, 1980.
- [5] S. K. Chang and T. L. Kunii. Pictorial data-base systems. *Computer*, 14:13–21, 1981.
- [6] Savvas A. Chatzichristofis, Yiannis S. Boutalis, and Mathias Lux. Img(rummager): An interactive content based image retrieval system. In *Proceedings of the 2009 Second International Workshop on Similarity Search and Applications*, SISAP '09, pages 151–153, Washington, DC, USA, 2009. IEEE Computer Society.
- [7] Savvas A. Chatzichristofis, Konstantinos Zagoris, Yiannis S. Boutalis, and Nikos Papamarkos. Accurate image retrieval based on compact composite descriptors and relevance feedback information. *IJPRAI*, 24(2):207–244, 2010.

- [8] Stéphane Clinchant, Julien Ah-Pine, and Gabriela Csurka. Semantic combination of textual and visual information in multimedia retrieval. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, pages 44:1–44:8, New York, NY, USA, 2011. ACM.
- [9] Charlie Dagli and Thomas S. Huang. A framework for grid-based image retrieval. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 2 - Volume 02*, ICPR '04, pages 1021–1024, Washington, DC, USA, 2004. IEEE Computer Society.
- [10] Lt.Dr.S.Santhosh Baboo Dr. K.Velmurugan. Content-based image retrieval using surf and colour moments. *Global Journal of Computer Science and Technology*, 11(10), 2011.
- [11] Re Xavier Falcao. Contour salience descriptors for effective image retrieval and analysis. *Image and Vision Computing*, 25:3–13, 2007.
- [12] Allen Gersho and Robert M. Gray. *Vector quantization and signal compression*. Kluwer Academic Publishers, Norwell, MA, USA, 1991.
- [13] Mark J. Huiskes and Michael S. Lew. The mir flickr retrieval evaluation. In *MIR '08: Proceedings of the 2008 ACM International Conference on Multimedia Information Retrieval*, New York, NY, USA, 2008. ACM.
- [14] Hervé Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In Andrew Zisserman David Forsyth, Philip Torr, editor, *European Conference on Computer Vision*, volume I of *LNCS*, pages 304–317. Springer, oct 2008.
- [15] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Improving bag of features for large scale image search. *International Journal of Computer Vision*, 87(3):316–336, 2010.

- [16] F Jurie and B Triggs. Creating efficient codebook for visual recognition. *ICCV*, pages 604–610, 2005.
- [17] Petrina A. S. Kimura, João M. B. Cavalcanti, Patricia Correia Saraiva, Ricardo da Silva Torres, and Marcos André Gonçalves. Evaluating retrieval effectiveness of descriptors for searching in large image databases. *JIDM*, 2(3):305–320, 2011.
- [18] Svetlana Lazebnik and Maxim Raginsky. Supervised learning of quantizer codebooks by information loss minimization. *IEEE PAMI*, 31:1294–1309, 2009.
- [19] Jia Li and James Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25:1075–1088, September 2003.
- [20] L. Liu, L. Wang, and X. Liu. In defense of soft-assignment coding. In *ICCV*, pages 1–8, 2011.
- [21] David G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, pages 1150–1157, 1999.
- [22] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.
- [23] Débora Myoupo, Adrian Popescu, Hervé Le Borgne, and Pierre-Alain Moëlllic. Multimodal image retrieval over a large database. In *Proceedings of the 10th international conference on Cross-language evaluation forum: multimedia experiments*, CLEF’09, pages 177–184, Berlin, Heidelberg, 2010. Springer-Verlag.
- [24] Mario A. Nascimento and Vishal Chitkara. Color-based image retrieval using binary signatures. In *Proceedings of the 2002 ACM symposium on Applied computing*, SAC ’02, pages 687–692, New York, NY, USA, 2002. ACM.
- [25] David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *Proceedings of the 2006 IEEE Computer Society Conference on Com-*

- puter Vision and Pattern Recognition - Volume 2, CVPR 06, pages 2161–2168, Washington, DC, USA, 2006. IEEE Computer Society.
- [26] Otavio Augusto Bizetto Penatti. Estudo comparativo de descritores para recuperação de imagens por conteúdo na web. *Universidade Estadual de Campinas, Instituto de Computação, Programa de pós-graduação (Dissertação de Mestrado)*, 2009.
- [27] Otávio Augusto Bizetto Penatti and Ricardo da Silva Torres. Eva: an evaluation tool for comparing descriptors in content-based image retrieval tasks. In *Multimedia Information Retrieval*, pages 413–416, 2010.
- [28] Matti Pietikäinen. Local binary patterns. *Scholarpedia*, 5(3):9775, 2010.
- [29] Gerard Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.
- [30] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.
- [31] Raimondo Schettini, Gianluigi CIOCCA, Silvia Zuffi, Istituto Tecnologie, and Infomatiche Multimediali. A survey of methods for colour image indexing and retrieval in image databases. In *In Color Imaging Science: Exploiting Digital*, pages 9–1. Media, John Wiley, 2001.
- [32] T. Sikora. The mpeg-7 visual standard for content description-an overview. *Circuits and Systems for Video Technology, IEEE Transactions on*, 11(6):696–702, jun 2001.
- [33] R. O. Stehling, Mario A. Nascimento, and A. X. Falcao. A compact and efficient image approach based on border/interior pixel classification. *CIKM*, pages 102–109, 2002.

- [34] Markus Stricker, Alexander Dimai, and Er Dimai. Color indexing with weak spatial constraints. In *in Proc. SPIE Storage and Retrieval for Image and Video Databases*, pages 29–40, 1996.
- [35] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J-M Geusebroek. Visual word ambiguity. *TPAMI*, 32:1271–1283, 2010.
- [36] N. S. Vassilieva. Content-based image retrieval methods. *Program. Comput. Softw.*, 35:158–180, May 2009.
- [37] James Z. Wang, Jia Li, and Gio Wiederhold. Simplicity: Semantics-sensitive integrated matching for picture libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:947–963, 2001.
- [38] Frank Wilcoxon. Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, 1(6):80–83, 1945.
- [39] Ka-Man Wong, Lai-Man Po, and Kwok-Wai Cheung. A compact and efficient color descriptor for image retrieval. In *ICME*, pages 611–614, 2007.
- [40] Ka-Man Wong, Lai-Man Po, and Kwok-Wai Cheung. Dominant color structure descriptor for image retrieval. In *ICIP (6)*, pages 365–368, 2007.
- [41] Dengsheng Zhang and Guojun Lu. Review of shape representation and description techniques. *Pattern Recognition*, 37(1):1 – 19, 2004.