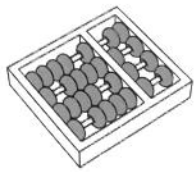


Felipe S. P. Andrade

O nome por extenso do aluno é "*Felipe dos Santos Pinto de Andrade*", conforme está expresso na ata da defesa.

Prof. Dr. Paulo Licio de Geus
Coord. de Pós-Graduação
Instituto de Computação - Unicamp
Matrícula 10.326-8

"Combinação de Descritores Locais e Globais para Recuperação de Imagens e Vídeos por Conteúdo"

Este exemplar corresponde à redação final da Tese/Dissertação devidamente corrigida e defendida por: Felipe dos Santos Pinto de Andrade
e aprovada pela Banca Examinadora.
Campinas, 25 de março de 2013

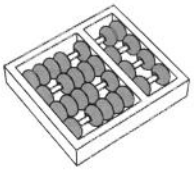
COORDENADOR DE PÓS-GRADUAÇÃO
CPG-IC

Prof. Dr. Paulo Licio de Geus
Coord. de Pós-Graduação
Instituto de Computação - Unicamp
Matrícula 10.326-8

CAMPINAS
2012

O nome por extenso do aluno é "*Felipe dos Santos Pinto de Andrade*", conforme está expresso na ata da defesa.

Prof. Dr. Paulo Licio de Geus
Coord. de Pós-Graduação
Instituto de Computação - Unicamp
Matrícula 10.326-8



Universidade Estadual de Campinas
Instituto de Computação

Felipe S. P. Andrade

"Combinação de Descritores Locais e Globais para Recuperação de Imagens e Vídeos por Conteúdo"

Orientador(a): **Prof. Dr. Ricardo da Silva Torres**
Co-Orientador(a): **Prof. Dr. Hélio Pedrini (Co-orientador)**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Computação da Universidade Estadual de Campinas para obtenção do título de Mestre em Ciência da Computação.

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO DEFENDIDA POR FELIPE S. P. ANDRADE, SOB ORIENTAÇÃO DE PROF. DR. RICARDO DA SILVA TORRES.

Ricardo Torres

Assinatura do Orientador(a)

CAMPINAS
2012

FICHA CATALOGRÁFICA ELABORADA POR
ANA REGINA MACHADO - CRB8/5467
BIBLIOTECA DO INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E
COMPUTAÇÃO CIENTÍFICA - UNICAMP

Andrade, Felipe dos Santos Pinto de, 1986-
An24c Combinação de descritores locais e globais para recuperação de imagens e vídeos por conteúdo / Felipe dos Santos Pinto de Andrade. – Campinas, SP : [s.n.], 2012.

Orientador: Ricardo da Silva Torres.
Coorientador: Hélio Pedrini.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Descritores. 2. Recuperação da informação. 3. Processamento de imagens - Técnicas digitais. 4. Programação genética (Computação). I. Torres, Ricardo da Silva,1977-. II. Pedrini, Hélio,1963-. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.

Informações para Biblioteca Digital

Título em inglês: Local and global descriptors combinations for content image and videos retrieval
Palavras-chave em inglês:
Descriptors
Information retrieval
Image processing - Digital techniques
Genetic programming (Computer science)
Área de concentração: Ciência da Computação
Titulação: Mestre em Ciência da Computação
Banca examinadora:
Ricardo da Silva Torres [Orientador]
Zenilton Kleber Gonçalves do Patrocínio Júnior
Anamaria Gomide
Data de defesa: 27-07-2012
Programa de Pós-Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

Dissertação Defendida e Aprovada em 27 de Julho de 2012, pela Banca examinadora composta pelos Professores Doutores:

Prof. Dr. Zenilton Kleber Gonçalves do Patrocínio Junior
ICEI / PUC-MG

Profª. Drª. Anamaria Gomide
IC / UNICAMP

Prof. Dr. Ricardo da Silva Torres
IC / UNICAMP

Instituto de Computação
Universidade Estadual de Campinas

Combinação de Descritores Locais e Globais para Recuperação de Imagens e Vídeos por Conteúdo

Felipe S. P. Andrade[1]

27 de Julho de 2012

O nome por extenso do aluno é "*Felipe dos Santos Pinto de Andrade*", conforme está expresso na ata da defesa.

Prof. Dr. Paulo Licio de Geus
Coord. de Pós-Graduação
Instituto de Computação - Unicamp
Matrícula 10.326-8

Banca Examinadora:

- Prof. Dr. Ricardo da Silva Torres (Orientador)
- Prof. Dr. Zenilton Kleber Gonçalves do Patrocínio Júnior (PUC-MG)
- Profa. Dra. Anamaria Gomide (IC/UNICAMP)
- Prof. Dr. João Paulo Papa (Suplente Externo - DC/UNESP)
- Prof. Dr. Anderson de Rezende Rocha (Suplente Interno - IC/UNICAMP)

[1]Apoio financeiro do CNPq (processo 135908/2009-4) no período de 2009–2011 e da *Advanced Micro Devices (AMD)* em 2012.

Abstract

Recently, fusion of descriptors has become a trend for improving the performance in image and video retrieval tasks. Descriptors can be global or local, depending on how they analyze visual content. Most of existing works have focused on the fusion of a single type of descriptor. Different from all of them, this work aims at analyzing the impact of combining global and local descriptors. Here, we perform a comparative study of different types of descriptors and all of their possible combinations. Furthermore, we investigate different models for extracting and comparing local and global features of images and videos, and evaluate the use of genetic programming as a suitable alternative for combining local and global descriptors. Extensive experiments following a rigorous experimental design show that global and local descriptors complement each other, such that, when combined, they outperform other combinations or single descriptors.

Resumo

Recentemente, a fusão de descritores tem sido usada para melhorar o desempenho de sistemas de busca em tarefas de recuperação de imagens e vídeos. Descritores podem ser globais ou locais, dependendo de como analisam o conteúdo visual. A maioria dos trabalhos existentes tem se concentrado na fusão de um tipo de descritor. Este trabalho objetiva analisar o impacto da combinação de descritores locais e globais. Realiza-se um estudo comparativo de diferentes tipos de descritores e todas suas possíveis combinações. Além disso, investigam-se modelos para extração e a comparação das características globais e locais para recuperação de imagens e vídeos e estuda-se a utilização da técnica de programação genética para combinar esses descritores. Experimentos extensivos baseados em um projeto experimental rigoroso mostram que descritores locais e globais complementam-se quando combinados. Além disso, esta combinação produz resultados superiores aos observados para outras combinações e ao uso dos descritores individualmente.

Agradecimentos

Gostaria de agradecer a todas as pessoas que contribuíram, direta ou indiretamente, para a realização deste trabalho. Em especial, gostaria de agradecer ao meu orientador, professor Ricardo da Silva Torres, por me ter me dado a oportunidade de fazer este trabalho. Agradeço desde o convite para realizar a iniciação científica em uma aula de estrutura de arquivos, fato que deu início a minha trajetória no meio acadêmico, até o último apoio, conselhos, correções e paciência durante as orientações em todos esses anos. Agradeço também ao professor Hélio Pedrini por aceitar a co-orientação e sua grande contribuição na orientação deste trabalho. Gostaria de agradecer aos meus pais por toda a criação que tive e pelo apoio que sempre me deram em todos os aspectos da vida. Finalmente, agradeço a todos os colegas de laboratório, de graduação e de outros lugares, tanto pela ajuda nos momentos de aprendizado e aplicações do conhecimento, quanto pela companhia nos momentos de lazer.

Sumário

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

Este capítulo descreve as principais motivações, objetivos e contribuições deste trabalho, assim como a organização do texto.

1.1 Caracterização do Problema

Ferramentas que lidam com conteúdo multimídia tornaram-se comuns atualmente devido ao fácil acesso à Internet com alto poder de transmissão de dados, aos equipamentos de produção e à redução dos custos de armazenamento. Em ferramentas de relacionamento social, por exemplo, o número de *uploads* diários de imagens e vídeos ultrapassa a casa dos milhões. O Youtube[1], *site* dedicado ao compartilhamento de vídeos, registra *upload* de mais de 48 horas de vídeo a cada minuto[2] e, em maio de 2010, excedeu uma taxa diária de mais de 2 bilhões de vídeos[3].

Neste cenário, buscar imagens, vídeos ou trechos de vídeos constitui uma das tarefas mais importantes demandadas nos últimos anos. Em geral, duas abordagens diferentes vêm sendo utilizadas para realizar a busca de tais dados: uma baseada em metadados textuais e outra baseada em busca por conteúdo.

A primeira abordagem consiste em associar metadados textuais a cada objeto e utilizar técnicas de busca em bancos de dados tradicionais para recuperação por palavra-chave. No entanto, esses sistemas requerem a anotação prévia dos objetos do banco de dados, o que é uma tarefa muito trabalhosa e, em geral, que demanda muito tempo. Além disso, o processo de anotação é muitas vezes ineficiente, pois os usuários normalmente não realizam anotações de forma sistemática. De fato, diferentes usuários tendem a utilizar diferentes expressões para descrever as mesmas características de um objeto. A falta

[1] `http://www.youtube.com` (último acesso em agosto de 2012).
[2] `http://www.youtube.com/t/faq` (último acesso em agosto de 2012).
[3] `http://www.youtube.com/t/press_timeline` (último acesso em agosto de 2012).

de sistematização no processo de anotação diminui o desempenho da busca baseada em palavras-chaves.

Diversas abordagens foram desenvolvidas para tratar essas limitações. Como exemplo, pode-se citar o uso de ontologias [10], anotação automática de imagens [48] e, em especial para este trabalho, o método conhecido como recuperação por conteúdo [26] (*Content-based Image Retrieval - CBIR*). Neste método, algoritmos de processamento de imagens e vídeos (geralmente automáticos) são utilizados para extrair características que representem propriedades visuais, como cor, textura, forma e relacionamento entre os objetos. Nessa abordagem, é possível recuperar dados similares a um padrão de consulta definido pelo usuário (imagem, quadro ou sequência de vídeo de consulta, por exemplo). Uma das principais vantagens desta técnica é a possibilidade de uma recuperação automática, em contraste ao esforço necessário para a anotação dos dados. A Figura 1.1 ilustra a realização de uma busca de imagens por conteúdo. Neste caso, busca-se por imagens similares àquela definida no topo.

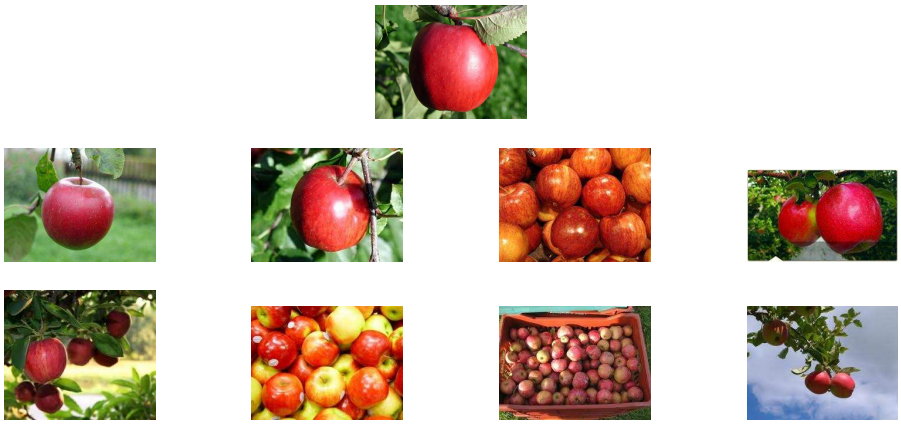


Figura 1.1: Exemplo de uma consulta e de resultados similares.

O descritor de imagens, definido por uma função de extração de características e uma função de medida de similaridade [7], é o responsável por caracterizar as propriedades visuais de imagens e vídeos. Duas imagens ou vídeos são considerados similares se a distância entre os vetores de características extraídos é pequena. Para extrair vetores de características, é possível tratar cada objeto multimídia de diversas maneiras, o que caracteriza diversos tipos de descritores. Uma dessas maneiras consiste em analisar as propriedades visuais globalmente, caracterizando os métodos utilizados como descritores *globais*. Os descritores *locais*, por outro lado, são computados sobre regiões entre fronteiras ou pontos de interesse [46].

Muitos sistemas de recuperação tendem a utilizar apenas características globais, que descrevem os dados como um todo ou características locais. Características

globais têm a habilidade de generalizar um objeto por completo como um único vetor. Consequentemente, seu uso em técnicas de classificação é simples. A Figura 1.2 ilustra o resultado de uma consulta no qual imagens são recuperadas segundo suas propriedades visuais globais.

Por outro lado, características locais são calculadas sobre vários pontos e são normalmente mais robustas às transformações que uma imagem pode sofrer, como rotações, variação de escala, oclusão de objetos e variações em iluminação. Por este motivo, há situações em que o uso de descritores locais permite a obtenção de resultados mais relevantes do que descritores globais. Por exemplo, considere os casos em que se deseja encontrar determinado objeto, mas imagens que contêm este objeto apresentam variações de iluminação, mudança de cor ou até mesmo oclusão, como ilustrado na Figura 1.3.

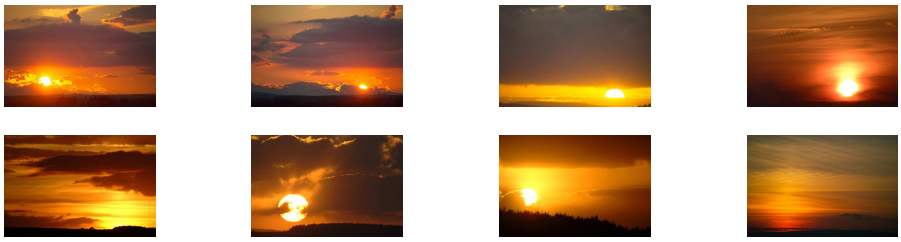


Figura 1.2: Exemplo de consulta em que propriedades *globais* são importantes para definir imagens de interesse.

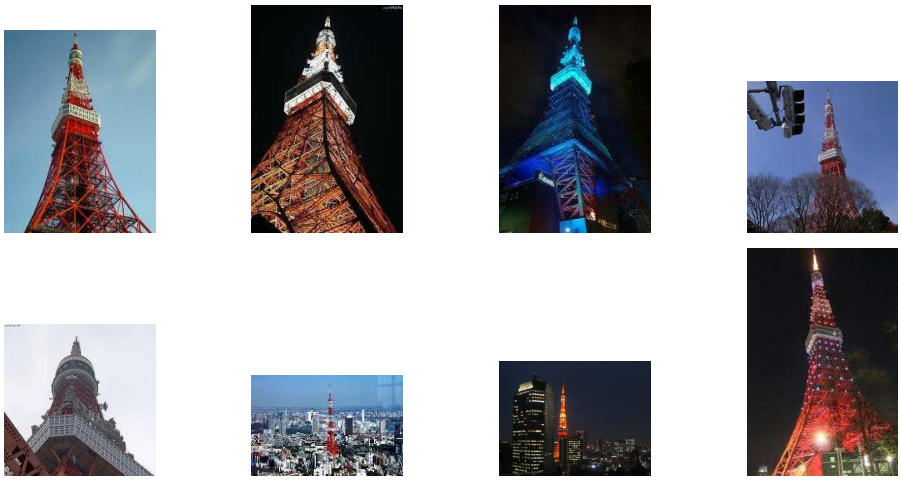


Figura 1.3: Exemplo de consulta em que propriedades *locais* são importantes para definir imagens de interesse.

Para melhorar o desempenho de sistemas de recuperação, uma nova tendência é a combinação de descritores. Muitos dos trabalhos existentes têm se concentrado na

combinação de apenas um tipo de característica (global ou local) [14, 39, 52]. Apesar de todos os avanços, ainda não é claro como a fusão de diferentes tipos de descritores afeta o desempenho destes sistemas, uma vez que existe a dificuldade de lidar com a diferença da natureza dos dados. Ao se utilizar diferentes tipos de propriedade visual para a realização de uma consulta, podem-se obter diferentes resultados. Neste caso, uma hipótese a ser investigada consiste em determinar se os resultados obtidos por descritores de diferente tipo (local e global) são complementares.

No domínio de recuperação, o uso de métodos de aprendizagem para combinar diferentes descritores [13] tenta aliviar o chamado problema de *gap* semântico, que consiste na dificuldade em se traduzir conceitos semânticos de alto nível presentes na imagem em propriedades de baixo nível (vetor de características). Uma dessas técnicas de aprendizagem é a Programação Genética (PG) [22]. Programação Genética é uma técnica da Inteligência Artificial para a solução de problemas baseados nos princípios da herança biológica e evolução. Nesse contexto, cada solução potencial é chamada de indivíduo em uma população. Sobre essa população são aplicadas transformações genéticas, como cruzamentos (*crossover*) e mutações, com o intuito de criar indivíduos mais aptos (melhores soluções) em gerações subsequentes. Uma função de adequação (*fitness*) é utilizada para atribuir valores para cada indivíduo com o objetivo de definir o seu grau de evolução, ou seja, quão bem soluciona um problema. O uso de PG para combinação de descritores pode ser descrito da seguinte forma: para um dado banco de imagens ou vídeos e um padrão de consulta, como uma imagem, o sistema retorna uma lista dos objetos que são mais similares às características da consulta, de acordo com um conjunto de propriedades da imagem representadas por descritores simples. Esses descritores são combinados utilizando-se um descritor composto em que a similaridade final é uma expressão matemática representada como uma árvore de expressão, em que os nós não-folhas são operadores numéricos e os nós folhas são um conjunto composto de valores de similaridade definidos por diferentes descritores [8]. Esse descritor composto gera uma nova lista que é avaliada pela função de *fitness* e, a partir do uso dos operadores genéticos, novos indivíduos são gerados de forma a buscar melhores resultados. Estudos recentes demostraram que a combinação de descritores globais utilizando PG apresentaram bons resultados para a recuperação de imagens por conteúdo [9, 13, 14].

1.2 Objetivos e Contribuições

O principal objetivo deste trabalho é investigar se a combinação dos descritores locais e globais no cenário de recuperação por conteúdo, tanto para imagens quanto para vídeos, produz resultados mais relevantes. Mais especificamente, busca-se responder as seguintes questões:

- A combinação de descritores globais e locais pode levar a resultados resultados mais relevantes?
- É possível utilizar programação genética para realizar esta combinação?
- É possível estender a estratégia de combinação baseada em programação genética para o problema de recuperação de vídeos?

As principais contribuições deste trabalho são:

- investigação de modelos para extração e a comparação das características globais e locais para recuperação de imagens e vídeos;
- a utilização da técnica de programação genética para combinar descritores globais e locais;
- uma análise abrangente dos resultados obtidos, respaldada em testes estatísticos. A partir dos experimentos realizados, pode-se observar que a combinação de descritores produz resultados superiores quando comparados ao uso dos descritores individualmente. Além disso, demonstra-se que a combinação de descritores locais e globais produz melhores resultados do que a combinação de descritores locais e a combinação de descritores globais.

1.3 Organização do Texto

Este texto está organizado da seguinte forma. O Capítulo 2 apresenta os principais conceitos e trabalhos relacionados. O Capítulo 3 descreve a metodologia proposta. Resultados experimentais são apresentados e discutidos no Capítulo 4. Finalmente, as conclusões e propostas para trabalhos futuros são apresentadas no Capítulo 5.

Capítulo 2

Conceitos e Trabalhos Relacionados

Neste capítulo, conceitos importantes para este trabalho, bem como trabalhos relacionados, são apresentados. A Seção 2.1 apresenta os fundamentos de imagens e vídeos. O conceito de recuperação de informação é apresentado na Seção 2.2. Os descritores de imagens utilizados no trabalho são apresentados na seção 2.3. Por fim, na Seção 2.4, a técnica de programação genética é apresentada.

2.1 Imagens e Vídeos Digitais

Uma imagem digital é a representação numérica da luz refletida em um determinado ponto representado no espaço. Dessa forma, uma imagem é definida como uma tupla $\hat{I} = (D_I, \overrightarrow{I})$, em que $D_I \subset Z^n$ é a posição de cada ponto amostrado, mapeado para o sistema de origem da imagem, e $\overrightarrow{I} = (I_1(p), I_2(p), ..., I_k(p))$ é uma função que mapeia cada pixel p da imagem em um valor real para todas as suas componentes. No caso de uma imagem em nível de cinza, a luz é amostrada em um ponto (x, y) pertencente ao $\mathbb{R}^2$ e quantizada para um valor inteiro. No caso de uma imagem colorida, a luz é quantizada para os valores de azul, verde e vermelho, tendo então três componentes. Cada elemento amostrado é considerado um pixel da imagem.

Um vídeo pode ser definido como uma coleção de imagens agrupadas sequencialmente conforme uma relação temporal [40]. Cada uma dessas imagens pode ser chamada de quadro (*frame*). Sendo assim, um vídeo de N quadros é definido como $V = (f_0, f_1, ..., f_{N-1})$, em que f_t representa o t-ésimo quadro do vídeo. Alguns desses quadros têm a característica de representar um intervalo contínuo do vídeo, que geralmente mantém uma mesma característica semântica e são chamados de quadros-chave.

2.2 Recuperação de Informação Visual por Conteúdo

Recuperação de informação visual por conteúdo é uma abordagem emergente que estende a recuperação de informação tradicional para repositórios de dados contendo informação visual, como imagens ou vídeos. Este método combina aspectos de processamento de sinais, visão computacional, aprendizado de máquina e recuperação de informação, possuindo aplicação em diversas áreas, tais como sistemas de busca na *Web*, vigilância, visualização e gerenciamento de dados [23].

Nas Seções 2.2.1 e 2.2.2, as técnicas de recuperação de imagens por conteúdo e recuperação de vídeo por conteúdo serão detalhadas em maior profundidade.

2.2.1 Recuperação de Imagens

Com o aumento de dados disponíveis em dispositivos de armazenamento e na rede, a busca por imagens se tornou uma tarefa importante. O modelo tradicional, a busca textual por palavras-chave, apresenta algumas limitações como as que foram apresentadas no Capítulo 1. Como alternativa a esta abordagem, surgiu a Recuperação de Imagens por Conteúdo (CBIR).

Os sistemas de CBIR têm, como tarefa, encontrar imagens visualmente parecidas com uma dada consulta. Nestes sistemas, surgiram diversos outros tipos de desafios. Como imagens são geralmente arquivos grandes, há a questão do armazenamento do grande volume de dados [41]. Por se tratar de conteúdo visual, a forma como os dados são apresentados é um ponto a ser avaliado [6]. Para a realização deste tipo de sistemas, as características das imagens são representadas por valores numéricos, tornando cada imagem um ponto em um espaço hiper-dimensional, levando a questão da dificuldade da indexação desses valores neste tipo de espaço (a chamada *maldição da dimensionalidade*) [50] e a questão da tradução do significado visual da imagem para os valores numéricos e sua possível não correspondência de informação (descontinuidade semântica).

A Figura 2.1 apresenta a arquitetura típica de um sistema de recuperação de imagens por conteúdo. A arquitetura é dividida em dois grandes módulos: inserção (linhas tracejadas) e processamento de consultas.

O subsistema da inserção de dados é responsável pela extração das características apropriadas das imagens e por armazená-las no banco de dados. Esse processo é geralmente feito de forma *offline*.

O processo de consulta, por sua vez, é organizado da seguinte forma: a interface permite que o usuário realize uma consulta a partir da definição de um padrão de consulta e visualize as imagens similares retornadas. O módulo de processamento de consultas extrai vetores de características do padrão de consulta e aplica uma função de distância

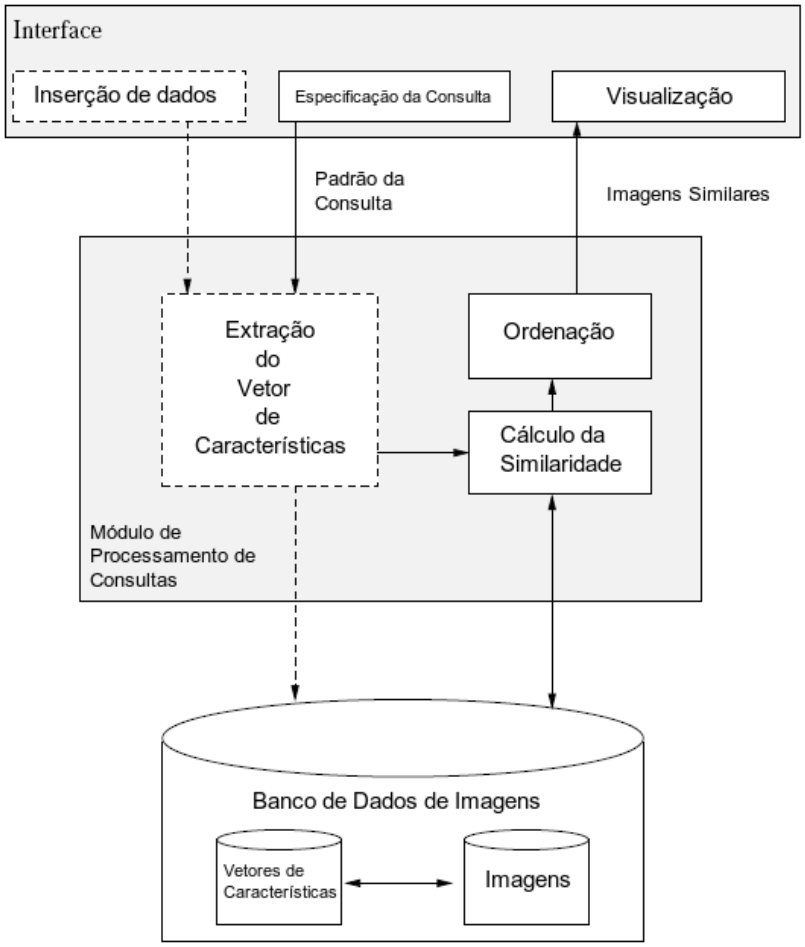


Figura 2.1: Arquitetura típica de um sistema de recuperação de imagens por conteúdo. (Adaptado de [7]).

(como a distância Euclidiana) para avaliar a similaridade entre a imagem de consulta e as imagens do banco de dados. Em seguida, uma ordenação das imagens do banco de imagens é realizada em ordem decrescente de similaridade à imagem consultada. Esses resultados são exibidos na interface.

2.2.2 Recuperação de Vídeos

Um vídeo contém múltiplos tipos de informação de áudio e visual, que são difíceis de extrair, combinar ou considerar na recuperação de vídeos por conteúdo em geral. Atualmente, os modelos de recuperação de vídeo por conteúdo são, em sua maioria, extensões diretas ou indiretas de técnicas de CBIR. Exemplos incluem escolher um quadro-chave de uma sequência significativa do vídeo e então extrair as características da imagem,

como cor ou textura, para indexação e recuperação, como pode ser visto em [54], que apresenta uma ferramenta de recuperação de vídeo que também integra recuperação textual. O sucesso de tal extensão, entretanto, pode ser prejudicado, uma vez que a relação espaço-temporal entre os quadros não é totalmente explorada [36]. De fato, características espaciais e temporais não têm sido empregadas corretamente na maioria dos sistemas de recuperação de vídeos por conteúdo, apesar da sua importância. Uma revisão de como a informação espaço-temporal pode ser melhor extraída, representada e comparada pode ser encontrada em [37].

Em geral, o cálculo de similaridade pode ser feito comparando-se o vídeo como uma sequência completa ou como um conjunto de subsequências. A comparação de subsequências requer o alinhamento e pareamento dos quadros ou quadros-chave ao longo do tempo. A comparação do vídeo como um todo, por outro lado, determina a similaridade entre duas sequências com base na distância das características representativas da sequência toda [17]. Para recuperar vídeos similares, além de computar a similaridade entre as sequências, a ordem de subsequências entre dois vídeos também pode ser levada em consideração. Meios mais sofisticados de medir a similaridade incluem comparação espaço-temporal [49] e comparação da linha de característica mais próxima [18].

Outras abordagens incluem, por exemplo, a detecção de objetos para realizar a recuperação. Em [25], um método de aprendizado não supervisionado é utilizado para extrair o objeto de interesse, permitindo a recuperação baseada em objetos. Após um objeto de interesse ser localizado em alguns quadros, um conjunto de características locais é extraído do objeto para representar o vídeo. Uma medida de similaridade foi proposta, de forma a computar a similaridade de um conjunto de vetores de características, determinando um *rank* em relação a uma consulta de uma só vez. Em [32], um método para indexação de vídeos em alta definição e ainda comprimidos, baseada em detecção de objetos, é apresentado. O primeiro passo consiste na extração do fundo, realizada por uma melhora da estimativa de movimentos na pirâmide *wavelet* JPEG, inicialmente proposta em [31]. O segundo passo consiste na definição das características do objeto, definidas por um histograma global do objeto, no domínio *wavelet*, em diferentes níveis de escalabilidade do espaço.

Para se inferir o contexto dos vídeos e realizar a recuperação, tem-se em [43] um método de mineração dos padrões temporais no conteúdo do vídeo, envolvendo a construção de duas árvores que funcionam como índices dos padrões encontrados nos vídeos, denominadas *fast-pattern-index tree* e *advanced fast-pattern-index-tree*, além de diferentes estratégias de busca nessa árvore na forma de dois algoritmos que utilizam o conceito de *re-rank*. Por fim, em [16], a recuperação dos vídeos é baseada em tomadas, na qual uma representação de fluxos ópticos é obtida a partir de *tensores*. Após uma redução do número de características utilizando-se LDA (*Linear Discriminant Analysis*) e PCA

(*Principal Component Analysis*), a recuperação é feita construindo-se *Hidden Markov Models* a partir da classificação das tomadas dos vídeos. Como mais de um modelo é construído, um para cada tipo de vídeo em diferentes níveis, forma-se uma árvore de Modelos de Markov. O processo de recuperação é feito então ao longo dessa árvore, do topo até o início e obtêm-se os resultados finais no último nível.

2.3 Descritores de Imagem

O descritor de imagens é responsável por quantificar o quão semelhante são duas imagens. Um descritor pode ser caracterizado por [7]: (i) algoritmo de extração de vetores de características das imagens e (ii) uma função de distância (medida de dissimilaridade).

Um modelo é apresentado na Figura 2.2, em que o descritor D recebe como entrada as imagens $\hat{I}_A$ e $\hat{I}_B$. A função de extração de características ϵ_D gera os vetores de características $\overrightarrow{v}_{\hat{I}_A}$ e $\overrightarrow{v}_{\hat{I}_B}$ que, por sua vez, servem de entrada para a função de distância δ_D, resultando na medida de similaridade d. A função do descritor é extrair informação de uma imagem a partir de características de baixo nível como cor, textura e forma.

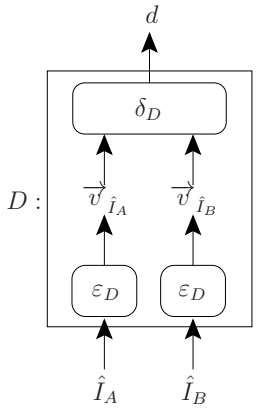


Figura 2.2: Uso de um descritor para computar a similaridade entre duas imagens. (Extraído de [7]).

Na Seção 2.3.1, uma introdução sobre medidas de similaridade é apresentada, enquanto nas Seções 2.3.2 e 2.3.3 são apresentados os métodos de descrições visuais e as medidas de similaridade utilizadas pelos descritores globais e locais, respectivamente.

2.3.1 Medidas de Similaridade

As medidas de similaridade, ou funções de distância, têm como objetivo determinar um valor que representa a proximidade dos vetores de características, ou seja, o quanto a

representação das imagens por esses vetores as considera similares. Como esta similaridade é representada por uma distância, o valor zero significa maior grau de proximidade, enquanto que valores maiores representam a dissimilaridade dos objetos.

Algumas medidas de similaridade são bem tradicionais como, por exemplo, a distância L1 (também chamada de *city-block* ou distância *Manhattan*). Dados dois vetores P e Q, sendo p_i q_i os i-ésimos pontos pertencetes à P e Q respectivamente, L1 é definido pela soma da diferença absoluta dos componentes dos vetores

$$L1(P,Q) = \sum_{i=1}^{n} |p_i - q_i| \tag{2.1}$$

Outro exemplo é a distância L2, ou distância Euclidiana, que representa a distância entre dois pontos no espaço, definida por

$$L2(P,Q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \tag{2.2}$$

No entanto, estas distâncias só podem ser utilizadas quando o número de características dos vetores é igual. Quando os vetores não possuem o mesmo número de características, há distâncias, como a *Earth Movers Distance* (EMD), que tratam destes casos. A ideia básica por trás da distância EMD é a seguinte: intuitivamente, dadas duas distribuições, uma pode ser vista como uma massa de terra espalhada no espaço e a outra como uma coleção de buracos no mesmo espaço. Dessa forma, a medida da EMD é definida como o mínimo esforço para preencher os buracos com a terra [38]. Ou seja, assumindo que as características de uma imagem são representadas por um conjunto de pontos em um espaço de dimensão $\mathbb{R}^d$, a distância entre dois conjuntos de pontos (representando duas imagens diferentes) é definida como o trabalho mínimo necessário para transformar um conjunto no outro. Formalmente, isto corresponde ao peso mínimo do casamento (ou fluxo) entre os dois conjuntos de pontos.

2.3.2 Descritores Globais

É possível tratar a imagem de diversas maneiras para extrair informações, o que caracteriza diversos tipos de descritores. Um estudo comparativo de diversos descritores pode ser encontrado em [34].

Uma dessas maneiras é considerar a informação de uma imagem globalmente, caracterizando os métodos utilizados por descritores globais. Como nenhum pré-processamento da imagem é necessário durante a extração de características, descritores que seguem esta abordagem normalmente apresentam algoritmos de extração de

características simples e rápidos. No entanto, descritores globais são pouco adequados para localizar detalhes, além de serem pouco robustos a deformações nas imagens. A Figura 2.3 ilustra uma representação de imagem em um vetor de características global.

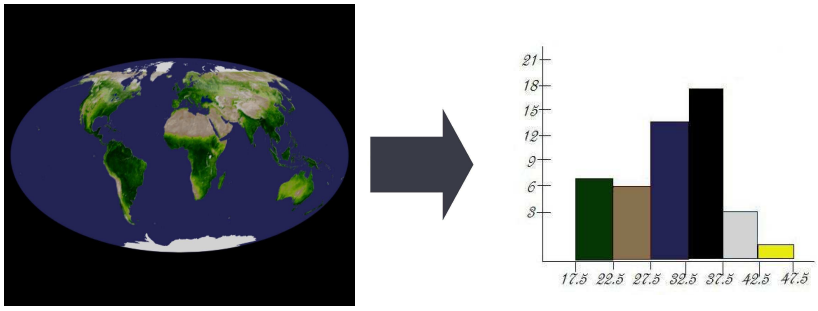


Figura 2.3: Representação de uma imagem em um vetor de características global.

A seguir, alguns dos descritores globais da literatura são apresentados.

Histograma Global de Cor

O histograma global de cor (GCH) é um dos primeiros descritores de imagem que utilizam a cor. Neste descritor, a informação de cor da imagem é representada em um histograma, tal que, dado um espaço de cor discreto definido por eixos de cor (como vermelho, verde, azul), o histograma é obtido ao se contar o número de vezes em que cada cor ocorre dentro da imagem. Histogramas são invariantes à rotação e translação sobre um eixo perpendicular ao plano da imagem e sofrem pouca alteração com mudança de ponto de vista, alteração na escala e oclusão [44].

A medida de similaridade utilizada no GCH é a distância Euclidiana. A utilização desta distância com histogramas de cor foi proposta em [29].

Border/Interior Pixel Classification

O *border/interior pixel classification* (BIC) [42] é um descritor projetado para lidar com imagens heterogêneas. Sua abordagem consiste em três etapas principais. A primeira é um algoritmo de análise de imagens que classifica os pixels da imagem em borda ou interior. A partir disso, com o espaço de cores RGB quantizado em 64 *bins*, dois histogramas são construídos, um para os pixels de borda e outro para os pixels de interior. A concatenação destes dois histogramas forma o vetor de características do BIC. Na última etapa, para realizar a medida de similaridade, utiliza-se a distância *dlog*. Esta função compara dois histogramas de acordo com uma escala logarítmica, diminuindo distorções nas distâncias geradas por *bins* com valores muito altos.

AutoCorrelograma de Cor

O AutoCorrelograma de Cor (ACC) [20] é o descritor que introduziu o conceito de correlograma de cor. As principais características desta abordagem são que ela inclui a correlação espacial entre as cores, pode ser usada pra descrever a distribuição global da correlação espacial local das cores, é fácil de computar e as características são pequenas.

Informalmente, um correlograma de cor de uma imagem é uma tabela, indexada por pares de cor, na qual a k-ésima entrada para $< i, j >$ define a probabilidade de se encontrar um pixel da cor j a uma distância k de um pixel da cor i. Para computar este vetor de características, o espaço de cor é quantizado em m cores. Após isso, é calculado um histograma de cor da imagem. Dado este histograma, é possível se determinar a probabilidade de um dado pixel ser de uma destas m cores.

Considerando que a imagem tem tamanho $n \times n$, uma distância máxima $d \in 1, 2, ..., n$ é escolhida a priori. Sejam i, j uma das m cores. Para cada distância $k \in 1, 2, ..., d$, uma tabela com a probabilidade de qualquer pixel da cor i a uma distância k de um pixel da cor j é criada. Dessa forma, considerando apenas pixels da mesma cor, é definida a correlação espacial destes pixels.

Este descritor foi proposto com uma variação da distância L1, na qual é levado em conta a diferença absoluta entre os valores, ou seja, a diferença recebe maior importância se está próxima dos valores iniciais.

Autocorrelograma Conjunto

A utilização de histogramas é baseada na suposição de que a distância entre os histogramas de duas imagens deve ser grande. No entanto, nem sempre isso ocorre na prática. Histogramas de cor são menos efetivos quando lidam com grandes coleções, pois imagens não relacionadas podem possuir histogramas similares. Para amenizar este problema, há o método de histograma conjunto (*joint histogram*), que utiliza outras características dos pixels em adição à cor como, por exemplo, densidade de arestas, texturização e magnitude de gradientes.

A técnica de correlogramas conjuntos é similar a de histogramas conjuntos. A diferença é que o correlograma inclui informação espacial sobre a imagem em adição à informação global das características. Correlogramas armazenam a probabilidade de se encontrar um pixel, dentro de um certo intervalo de valores para um conjunto de características, que está a uma certa distância de outro pixel com outro intervalo de valor para o mesmo conjunto de características. Nessa abordagem, a combinação de todas características possíveis é computacionalmente cara. Assim, na prática, apenas autocorrelogramas são utilizados. Autocorrelogramas utilizam apenas os pixels que possuem o mesmo conjunto de valores para um dado conjunto de características.

No autocorrelograma conjunto (*joint autocorrelogram* - JAC) [51], múltiplas características da imagem são utilizadas, entre elas, cor, textura, magnitude do gradiente e *rank*. O espaço de cor RGB é quantizado em um conjunto de valores discretos. Magnitude dos gradientes mede a taxa na qual a intensidade de um dado pixel varia na direção de uma mudança máxima, que pode ser calculada a partir da escala de cinza da imagem como o máximo da diferença dos vizinhos verticais e horizontais de um pixel. O *rank* quantifica a intensidade de variação na vizinhança de um pixel dentro da imagem.

A textura é definida como o número de vizinhos com nível de cinza maior, por uma certa quantidade que um pixel possui. A medida de similaridade é obtida a partir da diferença absoluta entre os valores de cada característica.

Espectro de Atividade Local

Outra característica muito utilizada das imagens é a sua textura. A textura de uma imagem pode ser definida como a caracterização da superfície de um objeto ou coleção de objetos.

Uma maneira efetiva de se descrever uma textura é a avaliação de bordas que, por sua vez, pode ser descrita por meio de um operador local apropriado para medir a variação espacial local ou "atividade", o gradiente. Assim, a distribuição do gradiente, que pode ser caracterizada por um histograma, pode ser utilizada para reconhecimento de textura. Tal histograma de gradiente é chamado de *gradient indexing*. Três operadores populares de gradiente são o Sobel, Prewitt e Roberts [33]. O histograma de gradientes é computado contando-se o número de pixels cujo valor de gradiente se enquadre em diversos *bins*.

Matematicamente, o gradiente mede a taxa de mudança em cada pixel, ou seja, a atividade local, entretanto, quando a operação de gradiente é aplicada a campos discretos, como uma imagem digital, e aproximada com o operador de gradiente, a saída representa apenas parte da atividade local. Com o objetivo de capturar a atividade espacial da textura ao longo das direções horizontal, vertical, diagonal e anti-diagonal separadamente, uma indexação de histogramas modificada, chamada de espectro de atividade local (*local activity spectrum* - LAS) [45], foi proposta na literatura. Sejam as quatro medidas de atividade no pixel (i, j) definidas por

$$
\begin{aligned}
g_1 &= |f(i-1, j-1) - f(i, j)| + |f(i, j) - f(i+1, j+1)| \\
g_2 &= |f(i-1, j) - f(i, j)| + |f(i, j) - f(i+1, j)| \\
g_3 &= |f(i-1, j+1) - f(i, j)| + |f(i, j) - f(i+1, j-1)| \\
g_4 &= |f(i, j+1) - f(i, j)| + |f(i, j) - f(i, j-1)|
\end{aligned}
$$

em que a distribuição de $[g_1 g_2 g_3 g_4]^T$ pode ser representada utilizando-se um histograma chamado de espectro de atividade local. Por utilizar tal medida, é possível diferenciar

texturas de acordo com suas atividades ao longo das quatro direções. Para realizar então o cálculo da similaridade entre os histogramas, utiliza-se a distância de quarteirão (*city-block*).

Histograma de Mudança Composto Quantizado

Utilizando uma ideia similar a de gradiente, outra maneira de caracterizar a textura é avaliar a variação média dos níveis de cinza de um pixel dentro de uma certa janela. Tal média é obtida da seguinte forma: sendo $y(i,j)$ o valor de cinza do pixel (i,j) e $N_r(i,j)$ um quadrado de vizinhos com raio r, com centro em (i,j), o nível de cinza médio em $N_r(i,j)$ é denotado por $y_r(i,j)$. Dessa forma

$$|y_r(i-r,j) - y_r(i+r,j)| = H_r^y(i,j)$$

define a medida da variação média de nível de cinza horizontal,

$$|y_r(i,j-r) - y_r(i,j+r)| = V_r^y(i,j)$$

define a medida da variação média de nível de cinza vertical,

$$|y_r(i-r,j-r) - y_r(i+r,j+r)| = D_r^y(i,j)$$

define a medida da variação média de nível de cinza diagonal,

$$|y_r(i+r,j-r) - y_r(i-r,j+r)| = A_r^y(i,j)$$

define a medida da variação média de nível de cinza anti-diagonal.

Utilizando $r = 1$, $H_1^y(i,j)$, $V_1^y(i,j)$, $D_1^y(i,j)$ e $A_1^y(i,j)$ descrevem a taxa de mudança dos níveis de cinza no pixel (i,j) em quatro direções diferentes em uma vizinhança 5×5.

Para descrever a taxa composta de mudança de nível de cinza do pixel (i,j), realiza-se a média aritmética das quatro direções

$$v(i,j) = (H_1^y(i,j) + V_1^y(i,j) + D_1^y(i,j) + A_1^y(i,j))/4$$

Para representar de forma eficiente o valor composto, uma quantização dos valores em 40 *bins* é realizada, definidos como $t(i,j)$. Calculando a distribuição da variação dos valores compostos quantizados, obtém-se o histograma de mudança composto quantizado (*quantized compound change histogram* - QCCH) [19]. A métrica L1 é utilizada no cálculo da medida de similaridade entre histogramas QCCH.

2.3.3 Descritores Locais

Os descritores locais são computados sobre características locais de imagens: regiões, fronteiras ou pontos de interesse. Repetibilidade é a qualidade mais importante para uma técnica que extrai características locais. Essa propriedade diz respeito ao fato de que uma imagem pode sofrer transformações geométricas ou radiométricas e, ainda assim, as mesmas características visuais são encontradas. Na prática, isso significa que as regiões, fronteiras ou pontos de interesse extraídos devem ter sofrido as mesmas transformações para recair sobre os mesmos objetos.

Pontos de interesse, que são pontos da imagem que podem ser univocamente localizados, são as características mais populares, devido a sua robustez. Uma vez que um ponto é detectado, uma pequena região em torno do ponto é utilizada para computar o descritor [46]. Há diversas abordagens de descritores locais de imagens e um estudo comparativo entre elas pode ser encontrado em [30]. A Figura 2.4 ilustra uma representação de imagem em um conjunto de vetores de características locais.

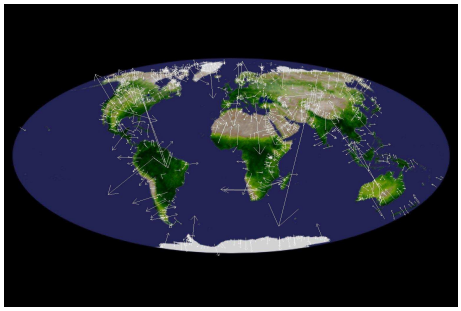


Figura 2.4: Representação de uma imagem em vetores de características locais.

A seguir, alguns dos descritores locais da literatura são apresentados.

Scale Invariant Feature Transform

Um descritor local robusto é o *Scale Invariant Feature Transform* (SIFT) [27]. Este descritor se tornou muito popular, pois transforma uma imagem em uma grande coleção de vetores de características locais, sendo cada um deles invariante à translação, escala e rotação da imagem, além de ser parcialmente invariante a mudanças de iluminação, projeções afins e projeções tridimensionais.

As características invariantes à escala são eficientemente identificadas por uma etapa prévia de filtragem. Esta primeira etapa identifica localizações importantes no espaço da escala procurando por regiões que são máximos ou mínimos da função de diferença de Gaussianas. Cada ponto é utilizado para gerar um vetor de características que descreve a região local da imagem amostrada relativamente à coordenada no quadrante do espaço

escalado. Esta descrição é feita por histograma de orientações de gaussiana (HoG). Os vetores de características resultantes são chamados de *SIFT keys*.

Speed Up Robust Features

Speed Up Robust Features (SURF) [3] é um descritor que se tornou popular por ser mais rápido e mantém a mesma acurácia que o SIFT. O detector deste descritor é baseado na matriz Hessiana. Para reduzir o tempo computacional, ele se baseia em imagens integrais, tal que o detector é conhecido como *Fast-Hessian*. Para determinar a escala e localização é utilizado o determinante da matriz Hessiana. O descritor, por outro lado, descreve a distribuição das respostas da transformada *wavelet* de Haar em uma vizinhança do ponto de interesse. Essas respostas são utilizadas primeiramente para obter uma orientação fixa dentro de uma região circular em torno do ponto de interesse. Após isso, é construída uma região quadrada alinhada à orientação selecionada e o vetor de características SURF é extraído.

Tais vetores são extraídos da região quadrada subdividida regularmente em 4 sub-regiões, formados pelo somatório das respostas dos coeficientes *wavelets* na direção horizontal d_x e vertical d_y, além do somatório dos valores absolutos das respostas $|d_x|$ e $|d_y|$ para agregar informação sobre a polaridade das mudanças de intensidade. Assim, cada sub-região possui um vetor de 4 dimensões $v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$, o que resulta em um vetor de características para todas as 4×4 sub-regiões de 64 dimensões, reduzindo o tempo de extração e de comparação das características.

Space-Time Interest Points

Ao se considerar vídeos, a componente temporal contém muita informação. Além disso, as estruturas de imagem em um vídeo não são restritas à velocidade constante e a acontecimentos constantes sobre o tempo. Pelo contrário, muitos eventos interessantes em vídeos são caracterizados pela forte variação nos dados, tanto na dimensão espacial quanto na temporal, por exemplo, em um cena com uma pessoa entrando em um quarto, ou pessoas aplaudindo, uma batida de carro ou água espirrando. Mais genericamente, pontos com uma movimentação não constante correspondem a estruturas locais da imagem em aceleração que, por sua vez, podem corresponder a objetos em aceleração no mundo real. Dessa forma, espera-se que tais pontos contenham informação sobre as forças atuando no ambiente físico e alterando estruturas.

Uma extensão dos pontos de interesse em imagem para o domínio epaço-temporal é denominada *Space-Time Interest Points* (STIP) [24]. Para detectar tais pontos no domínio espacial, o detector de Harris é tomado como base, que busca locais no espaço com alta variação nas direções. A partir disso, é utilizado um operador que detecta altas variações

na direção espacial e temporal. Pontos com tais propriedades corresponderão aos pontos de interesse espacial com locais distintos no tempo correspondendo às vizinhanças locais espaço-temporais com movimentação não constante.

Além de adaptar o operador de Harris para o espaço-tempo, também é adaptada a estimativa de escala local do operador Laplaciano para a combinação com o detector de Harris, derivando o detector Harris-Laplaciano invariante à escala. Para isso, os pontos escolhidos são o máximo da função de Harris adaptada, tanto no espaço quanto no tempo, como extremos do operador Laplaciano adaptado em diversas escalas espaciais e temporais. Isso é feito detectando-se pontos de interesse em um conjunto de escalas esparsamente distribuídos e então realizando o rastreamento destes pontos para o extremo da função Laplaciana adaptada.

Após a detecção dos pontos, para descrever a vizinhança espaço-temporal, é considerada uma derivada Gaussiana espaço-temporal computada nas escalas usadas para detectar os pontos de interesse correspondentes. A normalização em relação aos parâmetros de escala garante a normalização na variação de escala ao longo do tempo e espaço.

Histograma de Palavras Visuais

A categorização de textos é uma área muito estudada em recuperação de informação, na qual documentos são representados como *bag of words*, descritos como um vetor binário indicando a presença ou ausência de certos termos. Similarmente aos termos em um texto, uma imagem possui pontos de interesse locais, definidos como pequenas regiões que contêm rica informação local da imagem, geralmente associados a cantos e bordas.

Imagens podem ser representadas como conjuntos de vetores de características de cada ponto de interesse. No entanto, estes conjuntos variam em cardinalidade e são difíceis de se comparar, o que dificulta o processo de aprendizado. Para resolver este problema, é criado um agrupamento dos pontos para se obter um limite na dimensão das características. Cada grupo (*cluster*) é considerado uma palavra visual que representa um padrão local específico compartilhado pelos pontos de interesse naquele grupo. Assim, o processo de agrupamento gera um dicionário visual descrevendo os diferentes padrões locais das imagens [53]. Este dicionário pode variar o número de palavras (*clusters*), tornando-se mais ou menos refinado.

O número de *clusters* determina o tamanho do vocabulário, que pode variar de centenas a milhares de palavras. Mapeando os pontos de interesse para palavras visuais, pode-se representar cada imagem como *bag of visual words*. A representação do conjunto de palavras visuais pode ser convertida em um vetor de palavras visuais similar ao vetor de termos de um documento.

O vetor de palavras visuais pode ser composto pela presença ou ausência de cada

palavra visual na imagem, ou a contagem de cada palavra visual, ou ainda a contagem ponderada por outros fatores. A associação destas palavras com os pontos encontrados nas imagens pode ser de dois tipos [47]. Quando um ponto é associado unicamente a uma palavra, essa associação é considerada *hard*. Em contrapartida, um ponto pode estar associado a diversas palavras e é adicionado então uma valor v, de acordo com a distância relativa ao centroide de cada *cluster*, ponderada por uma Gaussiana. Estes valores podem ser definidos pelas seguintes equações:

$$v = \frac{1}{n}\sum_{i=1}^{n} K_s(D(w, p_i)) \tag{2.3}$$

em que n é o número de pontos de interesse em uma imagem, p_i é o ponto de interesse i, $D(w, p_i)$ é a distância entre a palavra visual w e o ponto de interesse p_i e σ é o parâmetro de suavização do kernel K, definido por:

$$K_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{1}{2}\frac{x^2}{\sigma^2}) \tag{2.4}$$

Após a associação das palavras, a contagem pode ser feita de diversas formas, por uma técnica chamada de *pooling* [5]. Quando a média do valor das palavras associada a cada ponto é utilizada para formar o vetor de características, tem-se a chamada *average pooling*. Outra possibilidade é a escolha da associação com o maior valor, denominada *max pooling*. Ao final dessa fase, o vetor corresponde a um histograma ponderado de palavras visuais e a similaridade entre eles pode então ser calculada com as métricas L1 ou L2.

Na Figura 2.5, pode-se ver que os pontos de interesse estão representados como pontos pretos na imagem. De cada um destes pontos, é extraído um vetor de características. Após esta extração, cada um destes pontos é associado a um *cluster* de palavras visuais do dicionário. Cada um destes *clusters* é um agrupamento de pontos segundo sua similaridade, definindo as palavras visuais W_1, W_2, W_3 e W_4.

2.4 Programação Genética

Programação genética é uma técnica da Inteligência Artificial para a solução de problemas baseados nos princípios da herança biológica e evolução. Nesse contexto, cada solução potencial é chamada de indivíduo em uma população. Um indivíduo é geralmente representado por uma árvore em que os nós folha representam os dados a serem manipulados e os nós internos representam qualquer tipo de operações que possam ser aplicadas nestes dados.

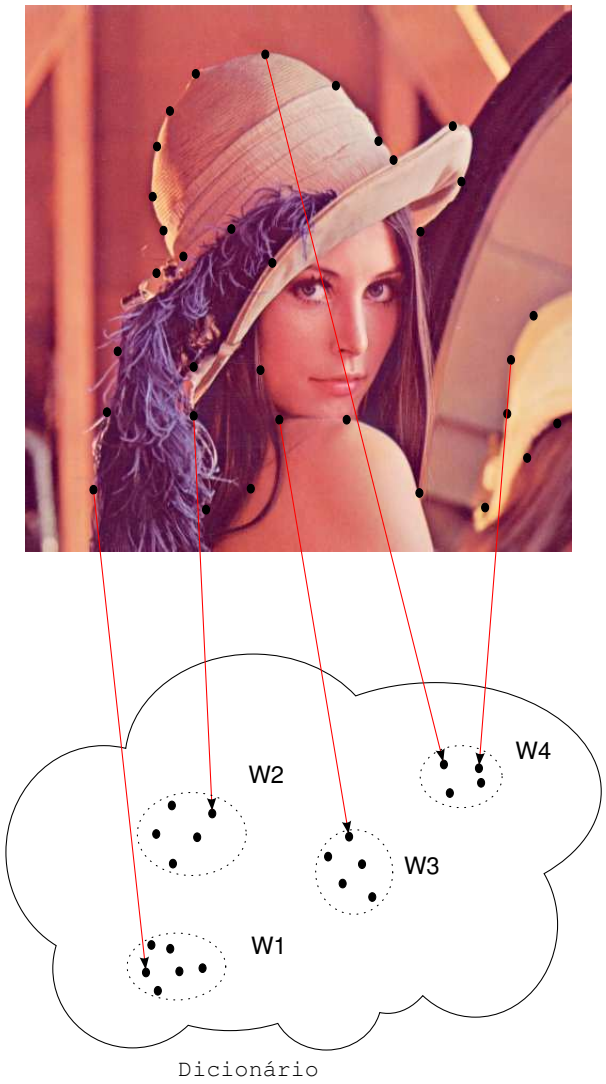


Figura 2.5: Atribuição de pontos a palavras de um dicionário visual.

Um exemplo de indivíduo pode ser encontrado na Figura 2.6. Nela, pode-se observar que os nós folhas são as variáveis x, y e z, e os nós internos são operadores matemáticos, resultando na função $x * z/y + \sqrt{x}$.

Dado esse modelo de indivíduos, uma população inicial é criada aleatoriamente. Para a criação desta população inicial, há três métodos principais. O primeiro é chamado de *grow*, no qual os indivíduos são gerados recursivamente a partir da raiz, no qual pode ser escolhida uma função ou um dado, até que apenas dados sejam escolhidos para os nós folhas de uma dada fase ou até atingir a altura máxima permitida. O segundo método, chamado de *full*, cria árvores do tamanho máximo, alocando apenas funções como nós internos, até atingir o último nível e escolher os dados para os nós folhas. Por fim, em

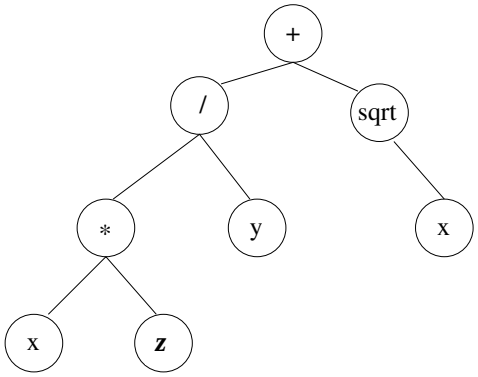


Figura 2.6: Representação de um indivíduo por meio da programação genética.

uma abordagem híbrida, metade da população é feita utilizando-se *grow* e a outra metade utilizando-se *full*. Esta abordagem é chamada de *half-and-half*.

Uma vez que a população inicial foi gerada, o processo de evolução tem início. O primeiro passo consiste em avaliar a qualidade de cada solução produzida por cada indivíduo. Este papel é realizado por uma função de adequação (*fitness*), em que, dado um conjunto de dados de treinamento, pode-se determinar o desempenho dos indivíduos neste conjunto e atribuir valores para cada indivíduo de modo a definir o seu grau de evolução. Após selecionar os melhores indivíduos, são aplicadas transformações genéticas sobre essa população com o intuito de criar indivíduos mais aptos (melhores soluções) em gerações subsequentes.

Os operadores genéticos utilizados são:

- *reprodução:* nesta operação, o indivíduo selecionado é copiado por completo para a nova população.
- *crossover:* nesta operação, as subárvores de dois indivíduos são selecionadas e é feita uma troca entre elas, gerando dois novos indivíduos para a próxima geração.
- *mutação:* nesta operação, após um indivíduo ser selecionado, é escolhida uma de suas subárvores, e outra subárvore é gerada aleatoriamente, que irá tomar o lugar dela, gerando um novo indivíduo para a população seguinte.

As operações de *crossover* e mutação são ilustradas nas Figuras 2.7 e 2.8, respectivamente. Na Figura 2.7, pode-se observar que os indivíduos na parte superior representam soluções escolhidas para uma dada geração. As subárvores destacadas em vermelho são então trocadas entre estes dois indivíduos, gerando novos indivíduos para uma geração posterior.

Na Figura 2.8, o indivíduo na parte superior, do lado esquerdo, tem uma subárvore selecionada, para ser trocada pela subárvore apresentada no lado direito, parte superior.

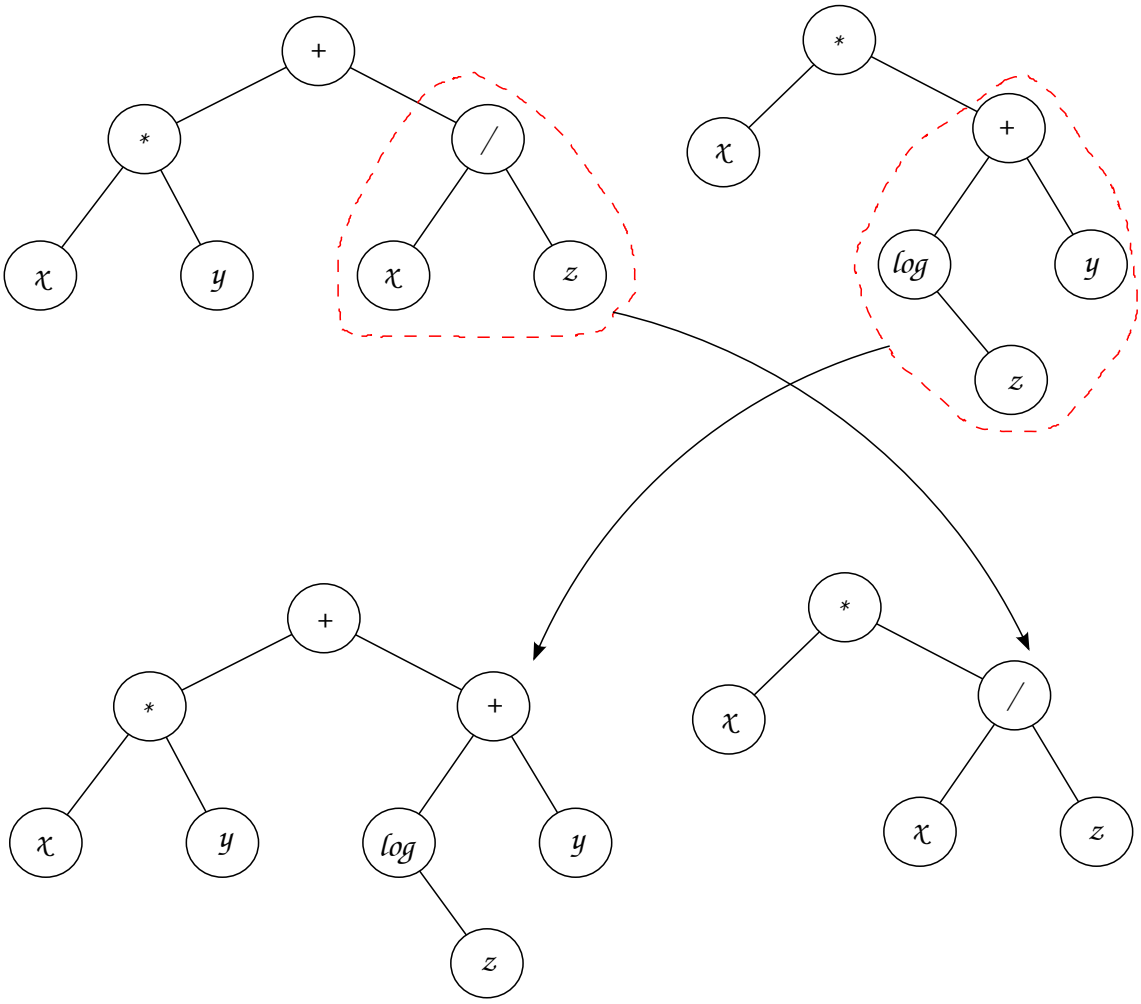


Figura 2.7: Operação de *crossover* entre indivíduos.

Após esta troca, um novo indivíduo é gerado para uma geração posterior. Este processo evolutivo está resumido no Algoritmo 1.

Um dos problemas enfrentados quando do uso de programação genética diz respeito ao fenômeno chamado de *overfitting*, em que as soluções encontradas ficam muito associadas ao conjunto de treinamento e acabam apresentando baixo desempenho no conjunto de testes. Para tratar tal problema é introduzido o conceito de conjunto de validação. Os indivíduos gerados são aplicados em um conjunto de validação e apenas os que se saírem bem em ambos os conjuntos serão aplicados ao conjunto de teste. O Algoritmo 2 apresenta o arcabouço de programação genética com conjunto de validação.

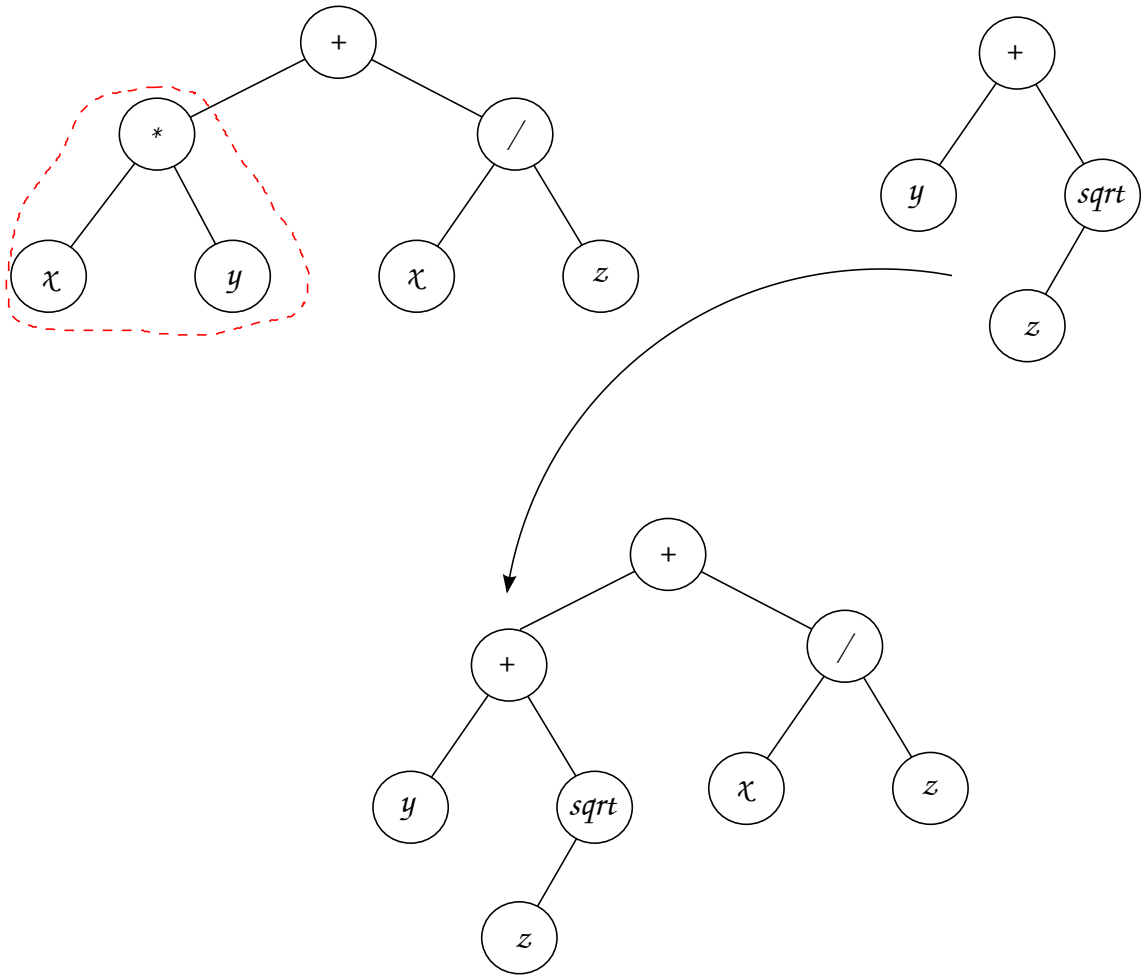


Figura 2.8: Operação de mutação em um indivíduo.

Algoritmo 1 Arcabouço de Programação Genética

1: Gerar uma população inicial aleatoriamente.
2: **while** número de gerações $\leq N_{gen}$ **do**
3: Calcular o *fitness* de cada indivíduo.
4: Selecionar os melhores N_{top} indivíduos.
5: **for all** os N_{top} indivíduos **do**
6: Aplicar reprodução.
7: Aplicar *crossover*.
8: Aplicar mutação.
9: **end for**
10: **end while**
11: Aplicação dos melhores indivíduos no conjunto de teste.

Algoritmo 2 Arcabouço de Programação Genética com Conjunto de Validação

1: Gerar uma população inicial aleatoriamente.
2: **while** número de gerações $\leq N_{gen}$ **do**
3: Calcular o *fitness* de cada indivíduo no conjunto de treinamento.
4: Selecionar os melhores N_{top} indivíduos.
5: **for all** os N_{top} indivíduos **do**
6: Aplicar reprodução.
7: Aplicar *crossover*.
8: Aplicar mutação.
9: **end for**
10: **end while**
11: **for all** os N_{top} indivíduos **do**
12: Calcular o *fitness* de cada individuo no conjunto de validação.
13: **end for**
14: Escolher o melhor indivíduo entre o conjunto de teste e validação.
15: Aplicação dos melhores indivíduos no conjunto de teste.

Capítulo 3

Metodologia

Um sistema de recuperação por conteúdo retorna uma lista de objetos de uma coleção, que são mais similares a um objeto de consulta, de acordo com o conteúdo visual. Para calcular a similaridade, é feita a combinação das características visuais dos objetos a partir da combinação dos seus descritores. Este processo pode ser dividido em dois grandes passos. Na Seção 3.1, o processo de descrição dos dados é apresentado. A segunda etapa, de combinação e aplicação, é apresentada na Seção 3.2.

3.1 Uso de Descritores

Para cada base, é feita a extração das características de todos os seus objetos, considerando todos os descritores definidos. Após esta etapa, é calculada a similaridade entre cada objeto para todos os outros pertencentes à mesma base.

Para as bases de imagens, no caso dos descritores globais, a extração de características gera um vetor por objeto e, portanto, o cálculo de similaridade é feito diretamente para cada par de vetores de características. No caso dos descritores locais, vários pontos de interesse são detectados por objeto, gerando múltiplos vetores de características por imagem.

Como cada objeto por ter um número diferente de vetores de características, o cálculo de similaridade entre eles foi efetuado utilizando duas abordagens. Na primeira delas, foi utilizado o conceito de incorporação da distância EMD, no qual os pontos são agrupados por *grids* de diferentes tamanhos e a distância é definida pelo histograma desses agrupamentos em diferentes níveis. Como a EMD não obedece o espaço Euclidiano, estruturas de dados para busca do vizinho mais próximo não podem ser utilizadas, dificultando o seu uso em grandes coleções de dados. Para isso, surgiu a ideia de se embutir a métrica em um espaço normalizado, ou seja, mapear cada ponto no espaço métrico em um ponto no espaço normalizado, para que a distância entre as imagens de

quaisquer pontos sejam comparadas à distância entre os pontos propriamente ditos [21].

A métrica EMD sofre uma baixa distorção para ser embutida em L_1^d (ou seja, $\mathbb{R}^d$ embutido com a norma L1), feita da seguinte forma: sendo P e Q dois conjuntos de pontos de cardinalidade s, cada um no $\mathbb{R}^k$ e $V = P \bigcup Q$, para cada par $p \in P, q \in Q$, o peso de (p, q) é a distância Euclidiana (L_2) entre p e q. Como a métrica EMD(P, Q) entre estes dois conjuntos é definida como o custo mínimo de casamento no grafo bipartido consistindo de todas as arestas entre pontos de P e Q, assumindo que a menor distância entre pontos é 1, e sendo $\triangle$ o diâmetro de V, grades (*grids*) são impostas no espaço $\mathbb{R}^k$ de lados $1/2, 1, 2, 4, ..., 2^i, ..., \triangle$. Sendo G_i uma grade de lado 2^i, é imposta a condição de que uma grade G_i é um refinamento da grade G_{i+i} e é transladada por um vetor escolhido aleatoriamente a partir de $[0, \triangle]^k$. Para cada grade G_i, é construído um vetor $v_i(P)$ com uma coordenada por célula, em que cada coordenada conta o número de pontos na célula correspondente. Em outras palavras, cada $v_i(P)$ forma um histograma de P. Um mapeamento f é então definido como um vetor f(P) que tem a forma $v_{-1}(P)/2, v_0(P), 2v_1(P), 4v_2(P), ..., 2^i v_i(P), ...$

Na segunda abordagem é feita uma clusterização dos pontos. Cada *cluster* definirá uma palavra visual. O conjunto de *clusters* irá então definir um dicionário de palavras visuais. Cada ponto será classificado de acordo com o dicionário obtido, obtendo-se assim um histograma de palavras para cada imagem. Neste caso, foi utilizado o *hard assignment* para associar os pontos das do dicionário aos de cada imagem.

No caso da base de vídeos, como cada objeto é composto por diversos quadros, a comparação entre eles é menos direta, pois cada vídeo possui um número variado de quadros. Para resolver este problema, a fase de extração de características foi feita da seguinte forma: primeiramente, foi escolhido um a cada 50 quadros dos vídeos para diminuir o tempo de processamento e a quantidade de dados armazenados. Essa escolha é possível, pois a informação visual relevante de interesse muitas vezes pode ser encontrada amostrando-se o vídeo. O valor de 50 quadros foi escolhido levando-se em conta que, em geral, os vídeos são exibidos a uma taxa de 24 quadros por segundo e, dessa forma, um quadro é amostrado aproximadamente a cada 2 segundos do vídeo, período em que poucas variações significativas podem ocorrer no vídeo.

No caso dos descritores globais, um vetor de características é extraído para cada quadro. Estes vetores são então agrupados no espaço de características, possibilitando-se o uso da aproximação L1 da função de distância EMD, visando ao cálculo de distância entre os vídeos. Esse processo pode ser visualizado na Figura 3.1.

No caso dos descritores locais, além do fato de existirem vários quadros por vídeo, cada quadro gera vários vetores de características. Após a extração de todos os pontos de todos os vídeos, é construído um dicionário de palavras visuais e palavras visuais são atribuídas aos pontos de cada quadro. Estes pontos podem ser atribuídos a mais de

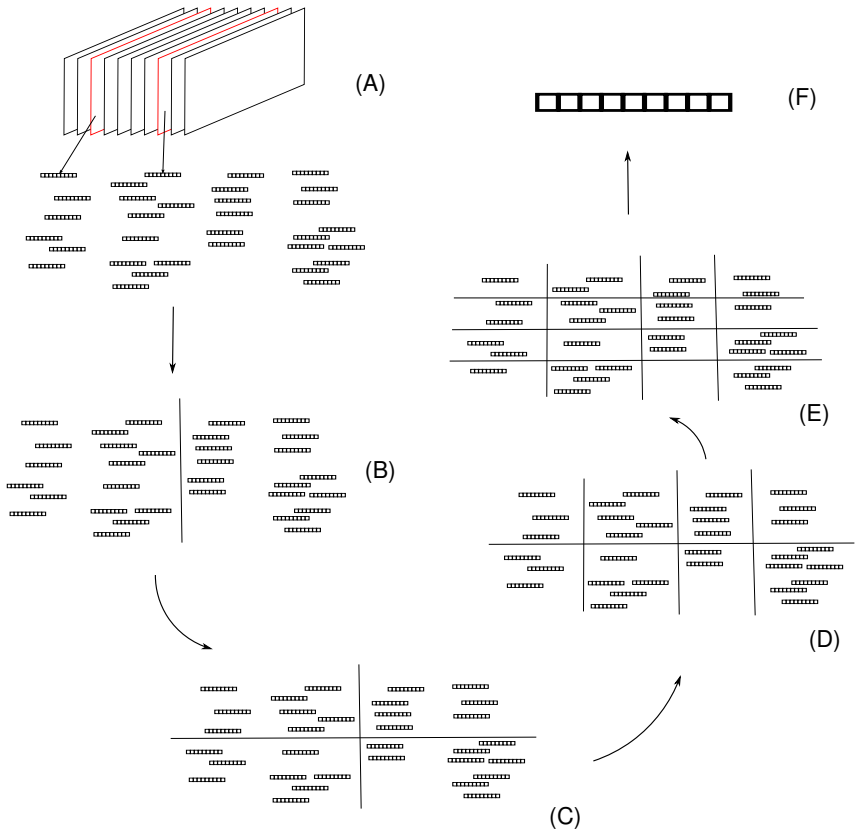


Figura 3.1: Extração das características globais dos quadros e agrupamento de vetores no espaço de características para cálculo de distância com a EMD embutida na métrica L1. No passo (A), são extraídos vetores de características dos quadros escolhidos. No passos de (B) a (E), os *grids* são estabelecidos. No passo (F), é definido o vetor de características do vídeo pela contagem das células dos *grids*.

uma palavra visual, de acordo com a distância relativa ao centroide do *cluster* da palavra, definidos pela equação 2.3. Uma vez atribuídos os pontos, o valor máximo de cada palavra é atribuído na posição correspondente do histograma (*max pooling*), para cada quadro. Esta última etapa é repetida considerando todos os quadros do vídeo, resultando em um único histograma. A Figura 3.2 ilustra este processo. Este processo também é aplicado em quadros amostrados para economizar processamento e armazenamento de dados.

3.2 Combinação de Descritores

Em [8], a técnica de programação genética é explorada para combinar descritores. O uso de GP para combinação de descritores pode ser definido da seguinte forma: para um dado banco de imagens e um padrão de consulta fornecido pelo usuário, como uma

imagem, o sistema retorna uma lista das imagens que são mais similares às características da consulta, de acordo com um conjunto de propriedades visuais do conteúdo.

Essas propriedades podem ser cor, textura ou forma e são representadas por descritores simples. Esses descritores são combinados usando um descritor composto DPG, em que a distância DPG é uma expressão matemática representada como uma árvore de expressão, em que os nós não-folhas consistem de operadores numéricos e os nós folhas são um conjunto composto de valores de similaridade definidos por diferentes descritores.

Esse processo está ilustrado na Figura 3.3. Seja $\mathcal{D} = \{D_1 = (\epsilon_1, \delta_1), D_2 = (\epsilon_2, \delta_2), D_3 = (\epsilon_3, \delta_3)\}$ um conjunto definido por três descritores. Primeiro, o algoritmo de extração ϵ_i é executado para cada objeto, então os vetores de características são concatenados. Em seguida, uma nova função de similaridade é obtida a partir da combinação das funções de similaridade δ_i, pelo GP. Esta nova função pode ser usada para computar a similaridade entre os objetos $\hat{I}_A$ e $\hat{I}_B$, utilizando a sua representação em vetores de características. Estes passos podem ser representados pela adaptação do Algoritmo 1, mostrada no Algoritmo 3.

Algoritmo 3 Arcabouço GP para combinação de descritores.

1: Gerar uma população inicial de "árvores de similaridades" aleatórias.
2: **while** numero de gerações $\leq N_{gen}$ **do**
3: Calcular o *fitness* de cada árvore de similaridade.
4: Gravar as primeiras N_{top} árvores de similaridade.
5: **for all** os N_{top} individuos **do**
6: Criar uma nova população a partir do uso das operações de reprodução, *crossover* e mutação.
7: **end for**
8: **end while**
9: Aplicar a melhor árvore de similaridade (a primeira árvore da última geração) em um conjunto de dados de teste.

Considerando os descritores GCH, BIC, LAS, SIFT com palvras visuais (siftBOF) e SURF com palavras visuais (surfBOF), um exemplo de indivíduo seria:

$$((0.2 * (las - gch))) + ((siftBOF * bic)/(surfBOF * gch)) \tag{3.1}$$

Na fase de treinamento, após a criação dos indivíduos de cada população, para cada imagem escolhida para ser usada como consulta no conjunto de treinamento, os objetos do conjunto de treinamento são ordenados utilizando-se a distância relativa à imagem de consulta, baseada no indivíduo em questão. Este *ranking* é avaliado por uma função de *fitness*, que levará em conta a quantidade de objetos relevantes retornados nas primeiras N posições. A função de *fitness* utilizada foi a função FFP4 [12], definida pela seguinte

equação

$$Fitness_{ffp4} = \sum_{i=1}^{|N|} r(d_i) \times k_8 \times k_9^i \tag{3.2}$$

em que $r(d) \in (0, 1)$ é a relevância associada a um documento, sendo 1 no caso de relevante e 0 caso contrário, e k_8 e k_9 são iguais a 7 e 0,982, respectivamente, tais como definidos em [12].

Após o cálculo do *fitness*, os indivíduos com maior pontuação são escolhidos para popularem as gerações seguintes por meio de replicações, mutações e operações de *crossover*. Após um determinado número de gerações, o indivíduo com maior pontuação é escolhido e aplicado para gerar ordenações no conjunto de testes.

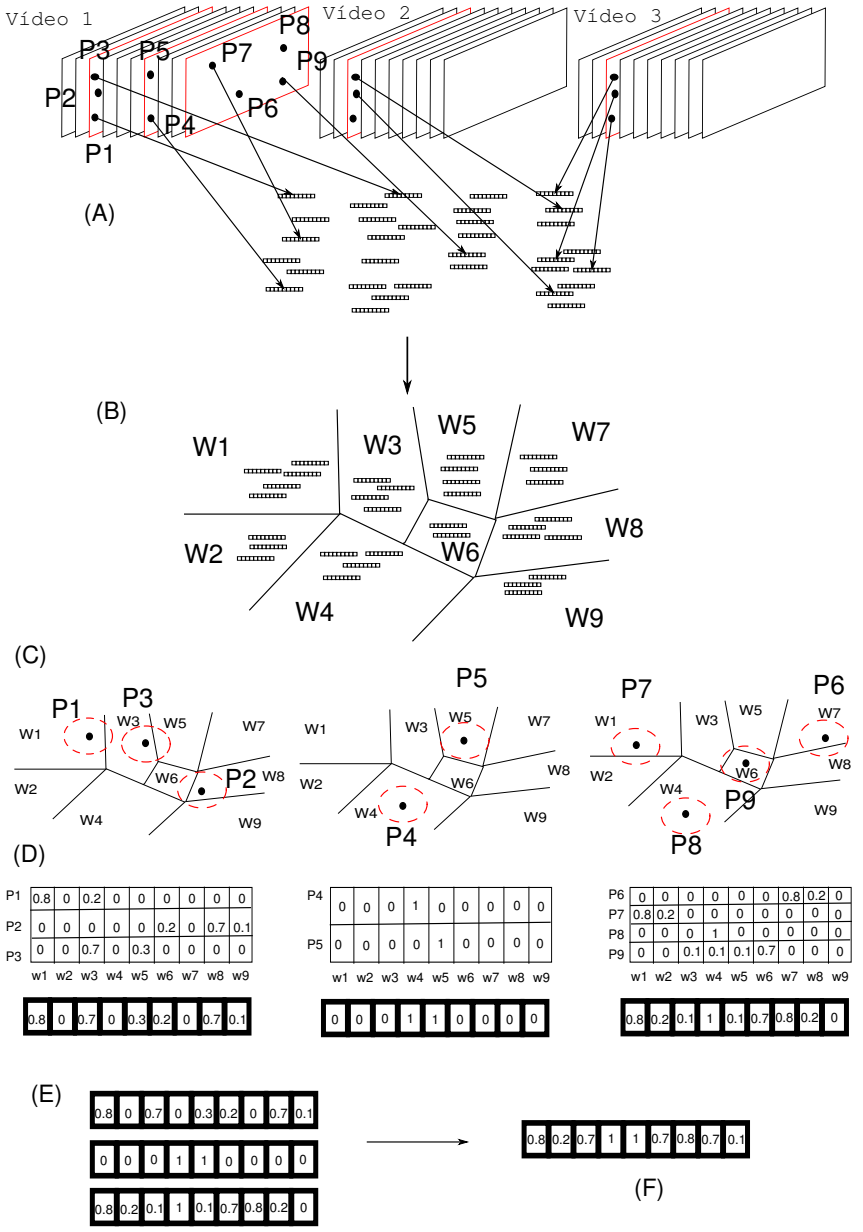


Figura 3.2: Extração das características locais dos quadros, construção do dicionário e associação com as palavras visuais. No passo (A), pontos de interesse são detectados e vetores de características são extraídos destes pontos nos quadros escolhidos. Este processo ocorre para todos os vídeos da base. Em (B), os vetores de características são separados em *clusters*, de acordo com a distância de um centroide. Cada região define uma palavra visual. Os pontos de interesse detectados são então associados às palavras visuais, de acordo com a proximidade de cada região, como pode ser visto em (C). O processo de *max pooling* é aplicado a todos os pontos de um mesmo quadro para determinar o máximo da ativação de cada palavra visual nos quadros. Este processo pode ser visto em (D), para os quadros escolhidos no vídeo 1. Finalmente, em (E), o processo de *max pooling* é aplicado aos quadros de um mesmo vídeo, resultando em apenas um vetor de características para o vídeo, apresentado em (F).

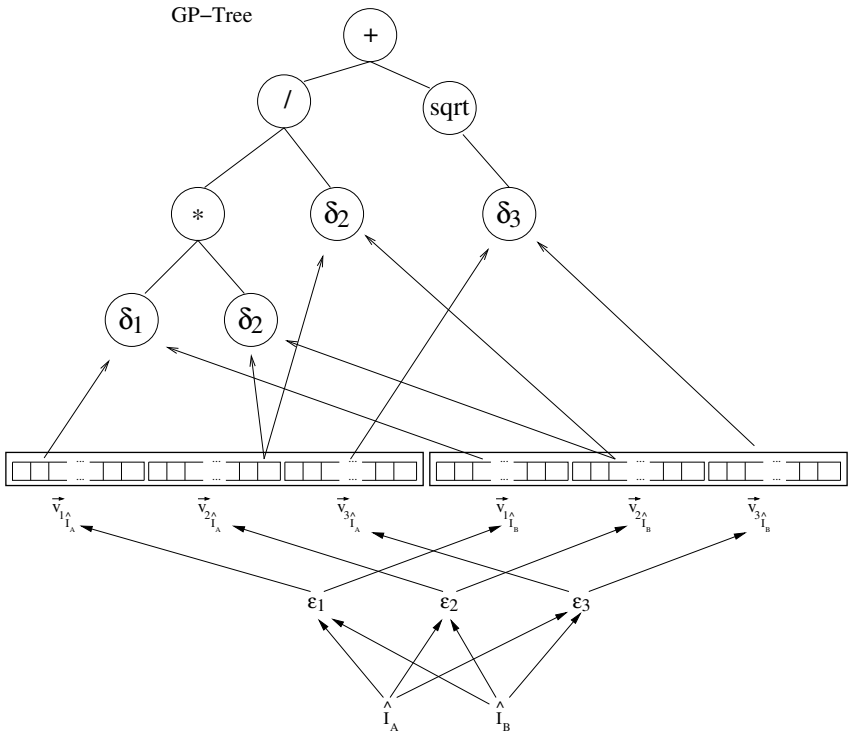


Figura 3.3: Função de similaridade GP.

Capítulo 4

Resultados Experimentais

Neste capítulo, o projeto experimental adotado é apresentado na Seção 4.1 e os resultados e as avaliações são apresentados na Seção 4.2.

4.1 Projeto Experimental

Uma vez definido o modelo de combinação, a sua avaliação foi realizada por meio do procedimento experimental *k-fold cross validation*. Na *k-fold cross validation*, o conjunto de dados inicial é separado aleatoriamente em k subconjuntos (denominados *folds*), nos quais não existem objetos repetidos entre eles, ou seja, os conjuntos são disjuntos. Um subconjunto é escolhido para ser usado como teste, enquanto os outros $k-1$ subconjuntos são usados como treinamento em uma técnica de aprendizado, nesse caso a PG.

O processo de validação cruzada é repetido k vezes, sendo que cada subconjunto é utilizado uma vez como teste. O resultado final é a média aritmética entre todos os subconjuntos. O principal objetivo deste método é testar todo o conjunto de dados e reduzir a variabilidade entre as rodadas. Nos experimentos descritos a seguir, o valor utilizado para k é 5. A seguir, detalhes das bases de dados utilizadas, configuração experimental e medidas de avaliação serão apresentados.

4.1.1 Descrição das Bases

A seguir, as bases utilizadas nos experimentos são descritas.

FreeFoto Nature

A base FreeFoto[1] é uma grande coleção de fotografias para uso não comercial disponível na Internet. Em nossos experimentos, foi utilizada uma parcela da base de dados intitulada como FreeFoto Nature. Nesta categoria, encontram-se imagens de cenas na natureza. Foram escolhidas 3461 imagens separadas em nove classes, que podem ser vistas na Tabela 4.1.

Classe	Número de imagens
Trees	496
Waves	853
Clouds	483
Flowers	204
Leaves	70
Sunset	616
Storm	83
Mountains	162
Waterfall	494

Tabela 4.1: Classes da base de imagens FreeFoto.

Caltech25

A Caltech[2] é uma base de imagens de objetos atualmente composta por mais de 30 mil imagens divididas em 256 categorias. De maneira similar à base FreeFoto, foram escolhidas apenas 25 categorias para serem utilizadas nos experimentos, totalizando 4991 imagens. As classes escolhidas podem ser vistas na Tabela 4.2.

Youtube10

A base de vídeos utilizada foi criada a partir de vídeos extraídos do site de compartilhamento YouTube, de modo a formar 10 categorias de vídeos, com um total de 696 vídeos utilizados, que em seu total possuem 245402 quadros. A Tabela 4.3 apresenta as classes escolhidas e o total de vídeos em cada uma delas.

4.1.2 Medidas de Avaliação

Foram utilizadas as seguintes medidas de avaliação: curva da precisão contra revocação, valores da Precisão contra 10 imagens retornadas (P@10), contra 5 imagens retornadas

[1] `http://www.freefoto.com`
[2] `http://www.vision.caltech.edu/Image_Datasets/Caltech101/`

Classe	Número de imagens
011.billiards	278
027.calculator	100
034.centipede	100
082.galaxy	81
086.golden-gate-bridge	80
096.hammock	285
099.harpsichord	80
105.horse	270
126.ladder	242
145.motorbikes-101	798
179.scorpion-101	80
183.sextant	100
186.skunk	81
197.speed-boat	100
200.stained-glass	100
201.starfish-101	81
204.sunflower-101	80
213.teddy-bear	101
218.tennis-racket	81
223.top-hat	80
232.t-shirt	358
233.tuning-fork	100
249.yo-yo	100
251.airplanes-101	800
253.faces-easy-101	435

Tabela 4.2: Classes da base de imagens Caltech25.

(P@5) e o valor da precisão média.

Uma das formas mais tradicionais de se avaliar a qualidade de um sistema de busca consiste em mensurar a qualidade da resposta das consultas. Do ponto de vista de um usuário, a maioria das pessoas desejaria que um sistema de recuperação funcionasse da seguinte forma: "recupere o máximo possível de itens relevantes e o mínimo possível de itens não relevantes". Simplificadamente, o primeiro critério corresponde ao conceito de revocação (R) e o segundo a noção de precisão (P).

Precisão e revocação são calculadas após um sistema determinar a ordem dos documentos pertencentes a uma coleção, em reposta a uma consulta. A precisão é então

Classe	Número de vídeos
Apple	77
Asparagus	64
Cat	66
Parrot	64
Cars	76
Soccer	63
Chile	73
WTC	73
JoSoares	64
OprahWinfrey	76

Tabela 4.3: Classes da base de vídeos Youtube10.

definida pela porção de documentos recuperados que são relevantes à consulta:

$$P = \frac{\#DRRet}{\#DRet}, \tag{4.1}$$

em que $\#DRRet$ é o número de documentos relevantes retornados e $\#DRet$ representa o número de documentos retornados. A revocação é a porção de documentos que são relevantes à consulta e foram retornados:

$$R = \frac{\#DRRet}{\#DRel} \tag{4.2}$$

em que $\#DRRet$ novamente representa o número de documentos relevantes retornados e $\#DRel$ representa o número de documentos relevantes.

Outra maneira de medir a eficácia é determinar a precisão obtida avaliando-se somente um certo número de imagens retornadas. Esta medida é chamada de precisão em N.

Os valores de precisão e revocação são obtidos do *rank* completo feito após uma determinada consulta. Levando em conta a ordem dos documentos nesse *rank*, é possível determinar uma curva de precisão contra revocação, calculando a precisão e a revocação em cada ponto do *rank* fazendo da precisão uma função da revocação, $P(R)$.

Uma vez obtidos os valores de precisão e revocação para diversos valores de documentos retornados, é possível calcular a precisão média sobre uma consulta, considerando o intervalo de revocação entre zero e 1. Este valor pode ser obtido pela integral da curva de precisão contra revocação, que na prática corresponderia ao somatório dos valores obtidos em todas as posições do *rank* [55]. Uma opção amplamente utilizada [11, 28] consiste em interpolar a função $P(R)$ determinando a precisão média sobre um conjunto de valores de revocação igualmente espaçados:

$$AveP = \frac{1}{11} \sum_{r \in 0,0.1,\ldots,1} P_{interp}(r) \tag{4.3}$$

em que a precisão interpolada $P_{interp}(r)$, significa que é escolhido o máximo da precisão de todos os pontos de revocação maiores que r. Uma vez obtida a precisão média para uma determinada consulta, pode-se calcular a média das precisões médias para todas as consultas realizadas, obtendo assim um único valor determinado por:

$$MAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q} \tag{4.4}$$

em que Q representa o número de consultas, $AveP(q)$ é a precisão média da consulta q em questão.

4.1.3 Descritores Utilizados

A seguir, são apresentados os descritores utilizados e, entre parênteses, a função de distância utilizada em cada um. A Tabela 4.4 apresenta os descritores globais utilizados, enquanto a Tabela 4.5 apresenta os descritores locais utilizados nos experimentos.

Descritor	**FreeFoto**	**Caltech25**	**Vídeos**
GCH (L2)	√	√	√
BIC (dlog)	√	√	√
JAC (L1)	√	√	
LAS (L1)	√	√	√
QCCH (L1)	√	√	√
ACC (L1)			√

Tabela 4.4: Descritores globais utilizados.

Nos experimentos em vídeos, o descritor JAC foi substituído pelo ACC, pois o JAC tem um vetor de características de 32000 *bins* e, como cada vídeo possui muitos quadros em que serão necessários realizar a extração de características, o tempo de processamento, o espaço de armazenamento e a aproximação com EMD passariam a ser requisitos custosos. Em função disso, foi escolhido o descritor ACC que possui um vetor de características menor e desempenho similar.

A distância EMD, indicada na Tabela 4.5, indica o uso da aproximação descrita na seção 3.1. Por simplicidade, os descritores SIFT e SURF com *bag of features*, será referido como siftBOF e surfBOF nas tabelas e graficos a seguir. L2 estará precedente ao nome do descritor quando esta distância for utilizada.

4.1.4 Parâmetros do GP

Os parâmetros utilizados para a evolução dos indivíduos são apresentados na Tabela 4.6. Estes parâmetros foram escolhidos de acordo com as análises feitas em [15]. Uma

Descritor	FreeFoto	Caltech25	Vídeos
SIFT (EMD)	√	√	
SURF (EMD)	√	√	
SIFT + Bag Of Features (L1)	√	√	√
SURF + Bag Of Features (L1)	√	√	√
SIFT + Bag Of Features (L2)			√
SURF + Bag Of Features (L2)			√

Tabela 4.5: Descritores locais utilizados.

investigação mais aprofundada do impacto dos parâmetros do arcabouço GP nos resultados de recuperação é deixada como trabalho futuro. A biblioteca de programação genética utilizada foi a *jgap*[3].

Parâmetro	Valor
População	30
Número de Gerações	10
Probabilidade de *Crossover*	0,8
Probabilidade de Mutação	0,2
Probabilidade de Reprodução	0,05
Função de *Fitness*	FFP4
Operadores	+,*,/,√, multiplicação por constante

Tabela 4.6: Parâmetros utilizados no GP.

4.2 Resultados

Esta seção apresenta os resultados da aplicação dos melhores indivíduos obtidos nas bases citadas. Os resultados para as bases de imagens estão na Seção 4.2.1 e os resultados com a base de vídeos estão na Seção 4.2.2.

4.2.1 Busca de Imagens

As Figuras 4.1, 4.2 e 4.3 mostram as curvas de precisão (*precision*) versus revocação (*recall*) para a base FreeFoto. Na Figura 4.1, pode-se observar que o descritor com melhor desempenho é o BIC e a combinação dos descritores globais supera todos os outros descritores ao longo da curva.

[3] `http://jgap.sourceforge.net/` (último acesso em agosto de 2012).

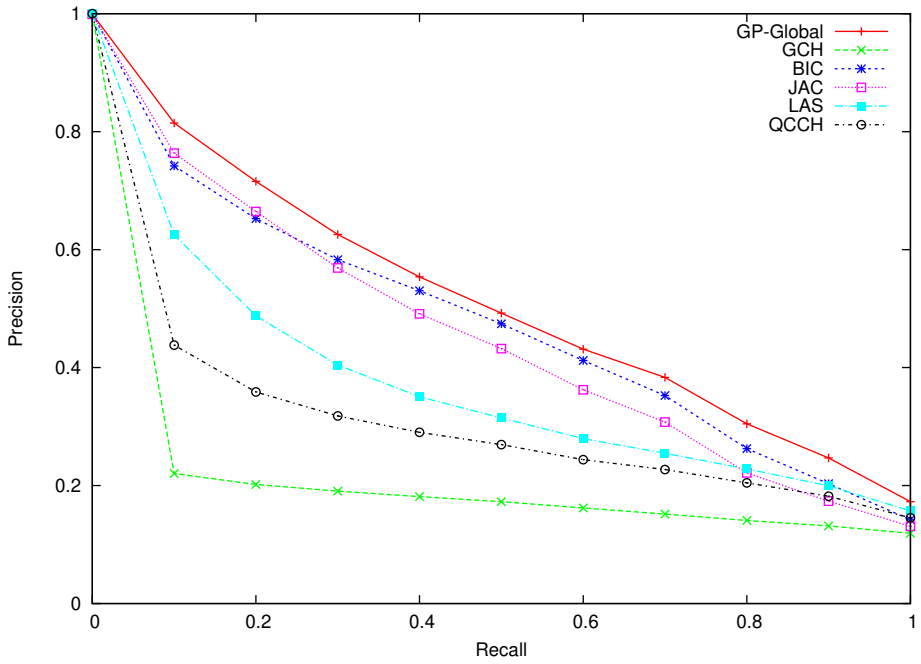


Figura 4.1: Curva de precisão versus revocação para a base FreeFoto (descritores globais).

Na Figura 4.2, o descritor com o melhor desempenho foi o SIFT utilizando histogramas de palavras visuais. A combinação dos descritores locais também superou os resultados individuais.

Quando todos os descritores foram combinados, como se pode observar na Figura 4.3, a curva da combinação apresenta os melhores resultados.

As médias de MAP para cada descritor e combinação podem ser vistas na Tabela 4.7. Da mesma forma que as curvas de precisão versus revocação, pode-se observar pelos valores de MAP que as combinações de descritores superam todos os outros descritores isoladamente e a combinação de todos os descritores possui os melhores resultados.

A Tabela 4.8 apresenta os valores de P@5 e P@10, em que se pode observar que o melhor descritor global é o JAC, enquanto o melhor descritor local é o SIFT utilizando palavras visuais. A combinação GP-GlobalLocal apresentou os melhores resultados.

As Figuras 4.4, 4.5 e 4.6 mostram as curvas de precisão versus revocação para a base Caltech25. Como a base Caltech25 possui mais imagens e mais classes, esta base pode ser considerada mais difícil do que a base FreeFoto Nature. A dificuldade inerente da base resultou em resultados piores quando as bases são comparadas. No entanto, pode-se observar o mesmo comportamento da base anterior. Na Figura 4.4, pode-se observar que todos os descritores têm resultados bem próximos, com o BIC sendo um pouco superior nas primeiras imagens retornadas. Em geral, a combinação destes descritores também se

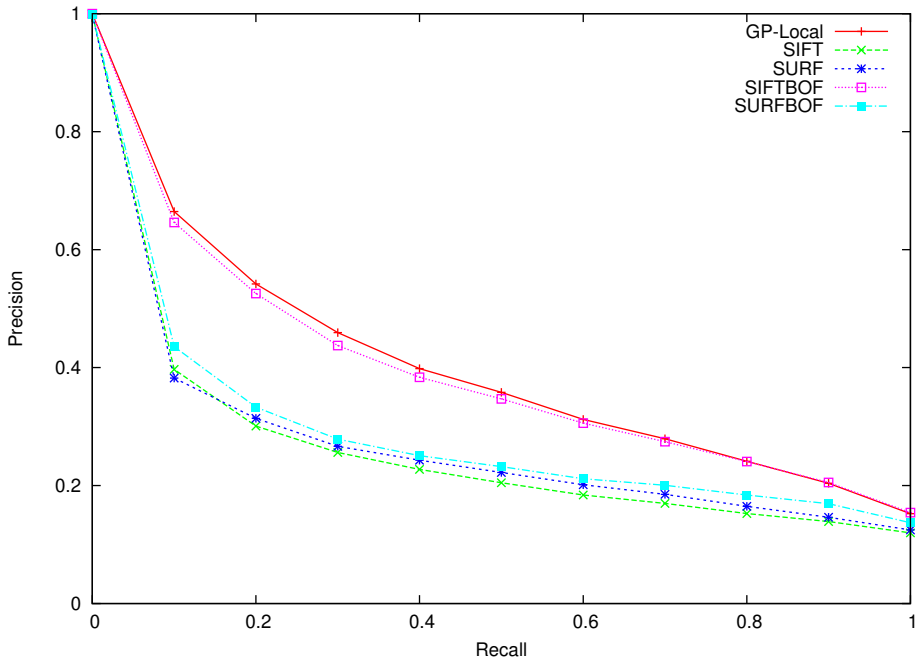


Figura 4.2: Curva de precisão versus revocação para a base FreeFoto (descritores locais).

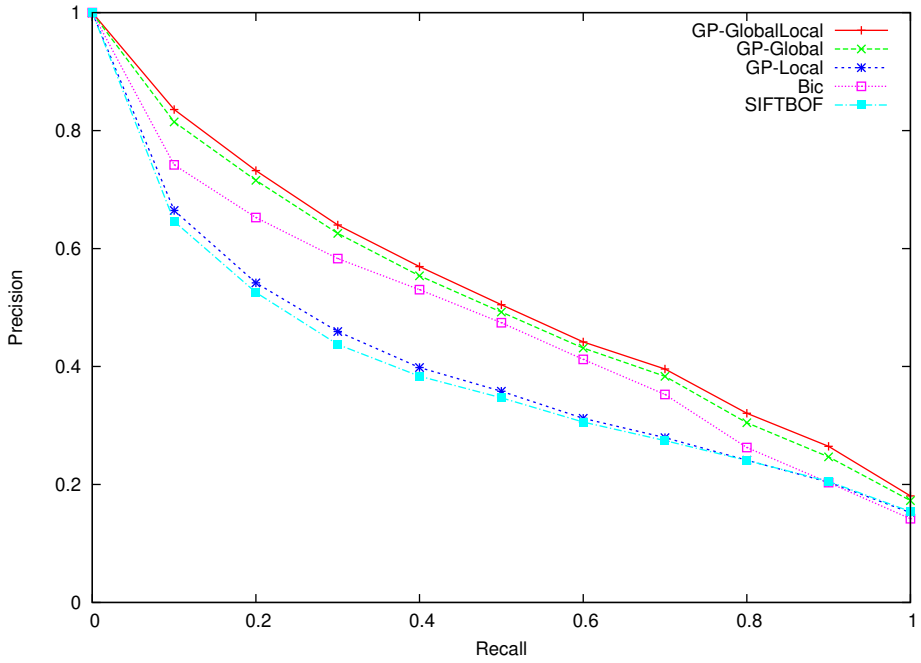


Figura 4.3: Curva de precisão versus revocação para a base FreeFoto (todas as combinações).

	Abordagem	MAP
Descritores Globais	GCH	0,15966
	BIC	0,45169
	JAC	0,43617
	LAS	0,35824
	QCCH	0,27859
Descritores Locais	SIFT	0,23961
	SURF	0,24774
	SIFTBOF	0,36783
	SURFBOF	0,26162
Combinações	GP-Global	0,49000
	GP-Local	0,38994
	GP-GlobalLocal	0,52112

Tabela 4.7: Resultados referentes à medida MAP para a base FreeFoto.

	Abordagem	P@5	P@10
Descritores Globais	GCH	0,1682	0,1665
	BIC	0,7668	0,6546
	JAC	0,7767	0,6735
	LAS	0,6196	0,5126
	QCCH	0,4608	0,3709
Descritores Locais	SIFT	0,2920	0,3461
	SURF	0,4247	0,3390
	SIFTBOF	0,6506	0,53994
	SURFBOF	0,4760	0,3679
Combinações	GP-Global	0,8191	0,71532
	GP-Local	0,6656	0,5522
	GP-GlobalLocal	0,8321	0,72762

Tabela 4.8: Resultados referentes às medidas P@5 e P@10 para a base FreeFoto.

mostrou superior.

Os descritores locais também apresentaram resultados próximos, sendo novamente o SIFT com o histograma de palavras visuais o descritor a apresentar os melhores resultados. A combinação destes descritores também obteve resultados ligeiramente superiores.

Analisando a combinação entre todos os descritores, pode-se verificar que os descritores locais apresentam os resultados mais baixos e a combinação de todos é superior a combinação utilizando apenas os descritores globais.

As médias de MAP para cada descritor e combinação podem ser vistas na Tabela 4.9.

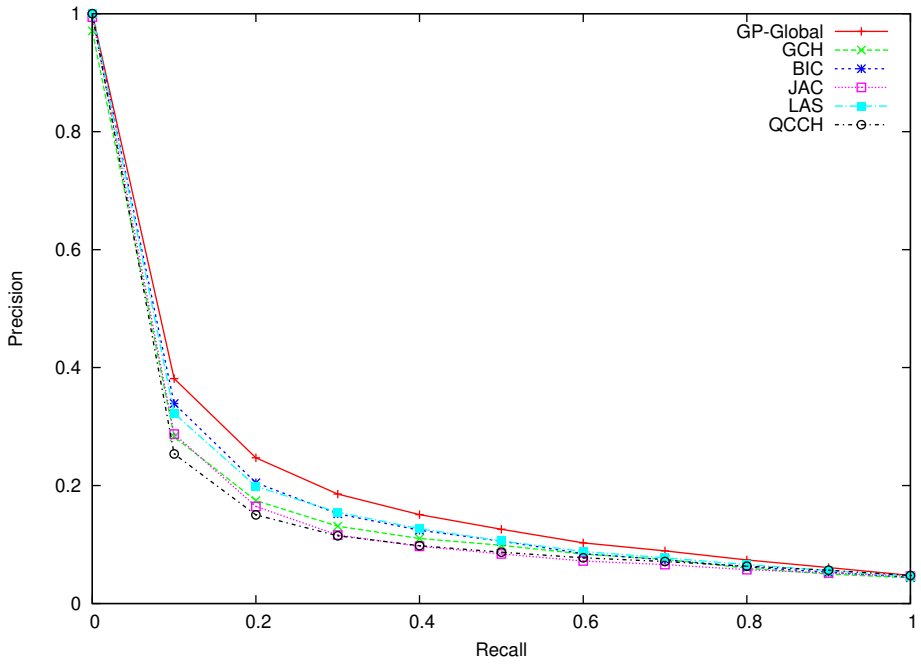


Figura 4.4: Curva de precisão versus revocação para a base Caltech25 (descritores globais).

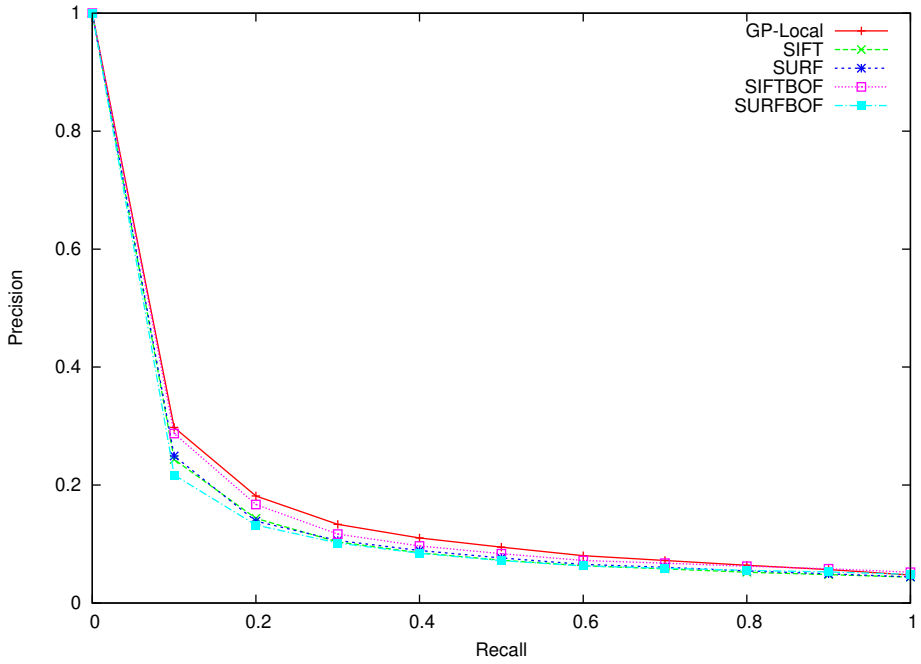


Figura 4.5: Curva de precisão versus revocação para a base Caltech25 (descritores locais).

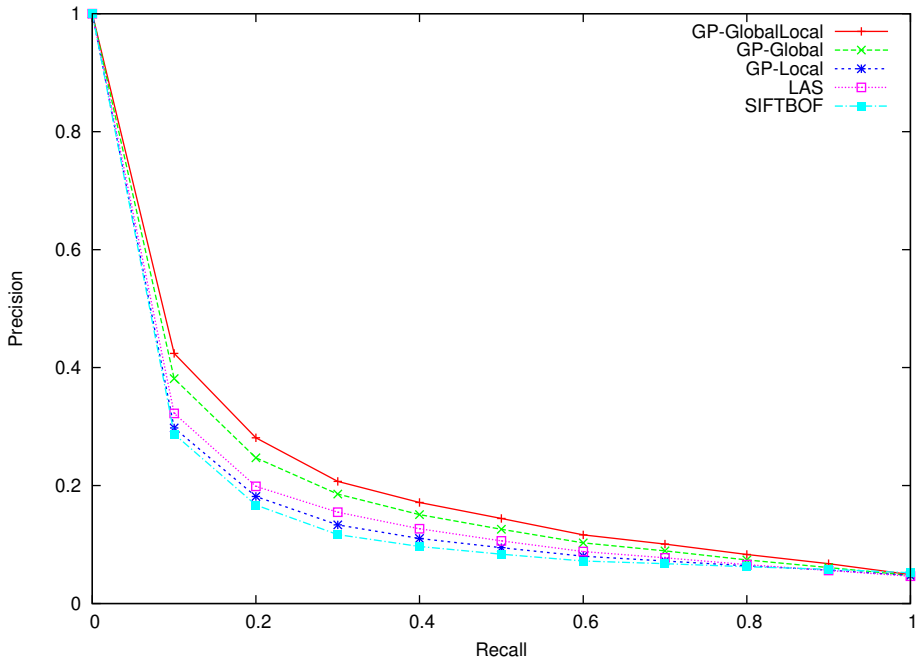


Figura 4.6: Curva de precisão versus revocação para a base Caltech25 (combinações).

Nela, pode-se verificar novamente que a combinação de todos descritores possui resultados melhores que as combinações globais e locais, que por sua vez possuem melhores resultados que os descritores isolados.

	Abordagem	**MAP**
Descritores Globais	GCH	0,15265
	BIC	0,16817
	JAC	0,14551
	LAS	0,17034
	QCCH	0,14690
Descritores Locais	SIFT	0,13504
	SURF	0,13607
	SIFTBOF	0,14353
	SURFBOF	0,12537
Combinações	GP-Global	0,19159
	GP-Local	0,15823
	GP-GlobalLocal	0,20858

Tabela 4.9: Resultados referentes à medida MAP para a base Caltech25.

Na Tabela 4.10, os resultados de P@5 e P@10 são apresentados. No caso dos descritores

globais, o descritor BIC apresenta o maior valor de P@5 e o LAS apresenta o maior valor de P@10.

	Abordagem	P@5	P@10
Descritores Globais	GCH	0,3249	0,2252
	BIC	0,3599	0,2491
	JAC	0,3082	0,2033
	LAS	0,3500	0,2538
	QCCH	0,3076	0,2099
Descritores Locais	SIFT	0,2920	0,1921
	SURF	0,2870	0,1885
	SIFTBOF	0,2950	0,1891
	SURFBOF	0,2553	0,1551
Combinações	GP-Global	0,3838	0,2811
	GP-Local	0,3324	0,2289
	GP-GlobalLocal	0,4051	0,2972

Tabela 4.10: Resultados referentes às medidas P@5 e P@10 para a base Caltech25.

4.2.2 Busca de Vídeos

As Figuras 4.7, 4.8 e 4.9 mostram as curvas de precisão versus revocação para a base de vídeos do Youtube. Na Figura 4.7, pode-se observar que os descritores LAS e QCCH tiveram resultados praticamente sememlhantes, com leve vantagem para o LAS em valores de revocação maior. Em valores de revocação menor, BIC e GCH se mantinham empatados, enquanto a combinação GP-Global se mantinha superior. Conforme o valor de revocação vai aumentando, pode-se observar que as três curvas vão se aproximando, com superação do GCH no final.

Na Figura 4.8, pode-se observar que os descritores baseados em SURF obtiveram os piores resultados, enquanto houve um empate entre os descritores baseados em SIFT e a combinação GP-Local.

Finalmente, comparando-se as combinações com os melhores resultados, pode-se observar que a combinação GP-GlobalLocal obteve os melhores resultados.

As médias de MAP para cada descritor e combinação podem ser vistas na Tabela 4.11. A Tabela 4.12 apresenta os valores de P@5 e P@10. O melhor descritor global foi o GCH. O SIFT com palavras visuais e distância L2 é o melhor descritor local. Novamente, a combinação GP-GlobalLocal apresentou os melhores resultados.

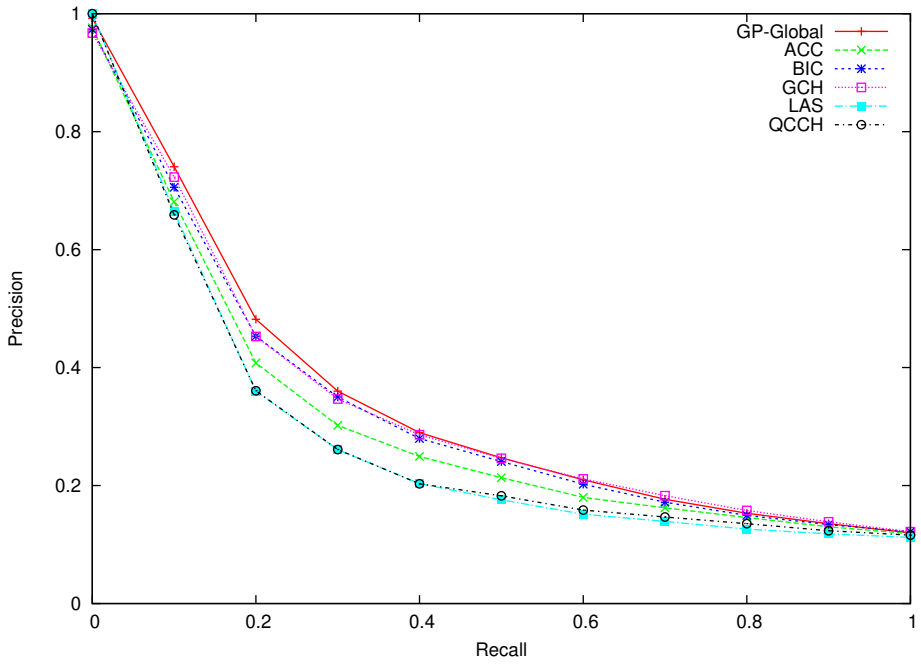


Figura 4.7: Curva de precisão versus revocação para a base Youtube10 (descritores globais).

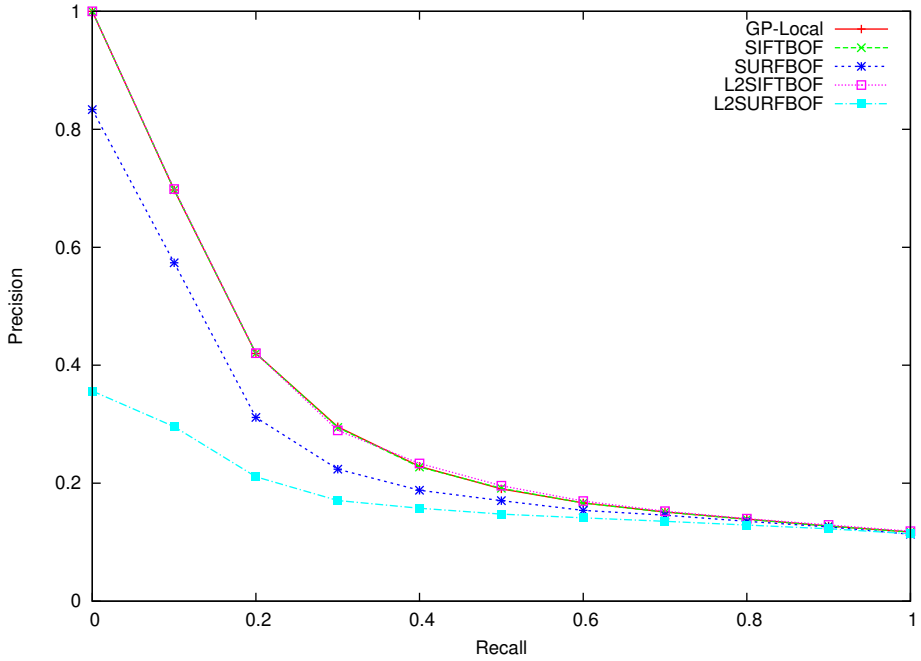


Figura 4.8: Curva de precisão versus revocação para a base Youtube10 (descritores locais).

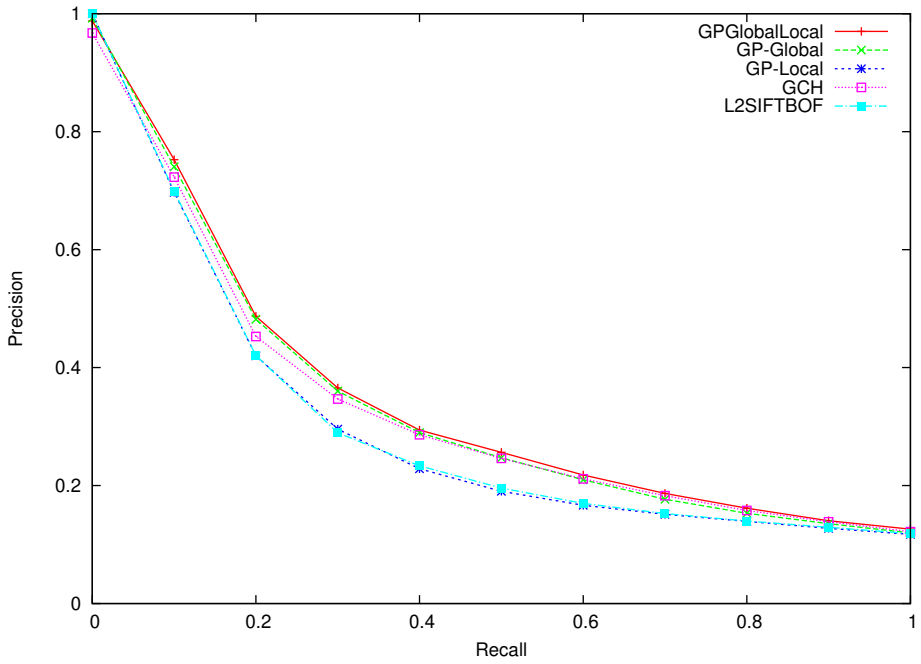


Figura 4.9: Curva de precisão versus revocação para a base Youtube10 (combinações).

	Descritor	MAP
Descritores Globais	GCH	0,31464
	BIC	0,30928
	ACC	0,2894
	LAS	0,26353
	QCCH	0,26736
Descritores Locais	SIFTBOF	0,2852
	SURFBOF	0,23580
	L2SIFTBOF	0,28664
	L2SURFBOF	0,15475
Combinações	GP-Global	0,32155
	GP-Local	0,28543
	GP-GlobalLocal	0,32799

Tabela 4.11: Resultados referentes à medida MAP para a base Youtube10.

4.3 Discussão dos Resultados

Por meio dos experimentos, pode ser observado que as funções de similaridade baseadas em GP têm um desempenho melhor que todos os descritores. O GP-Global tem um resultado

	Descritor	P@5	P@10
Descritores Globais	GCH	0,3934	0,2754
	BIC	0,3902	0,2702
	ACC	0,3575	0,2497
	LAS	0,3180	0,2216
	QCCH	0,3224	0,2204
Descritores Locais	SIFTBOF	0,3534	0,2396
	SURFBOF	0,2965	0,2063
	L2SIFTBOF	0,3544	0,2391
	L2SURFBOF	0,1262	0,1204
Combinações	GP-Global	0,4023	0,2765
	GP-Local	0,3523	0,2400
	GP-GlobalLocal	0,4143	0,2800

Tabela 4.12: Resultados referentes às medidas P@5 e P@10 para base Youtube10.

ligeiramente maior do que os descritores globais. No entanto, o mesmo não ocorre para o GP-Local, que não melhorou significativamente a efetividade dos descritores locais. Por outro lado, a combinação dos descritores globais e locais é promissora, apresentando os melhores resultados, como pode ser verificado pela superioridade da combinação GP-GlobalLocal em comparação aos descritores isolados.

Como os resultados apresentam pouca diferença entre si, o teste T-pareado foi realizado para demonstrar a significância estatística destes resultados. Para isso, os intervalos de confiança entre as diferenças das médias de cada classe foram computadas, para comparar cada par de abordagem. Se o intervalo de confiança incluir zero, a diferença não é significativa, com aquele nível de confiança. Se o intervalo de confiança não incluir zero, então o sinal do intervalo indica qual alternativa é melhor.

As Tabelas 4.13 e 4.14 apresentam os intervalos de confiança (com confiança de 95%) das diferenças entre os métodos baseados em GP e os melhores descritores globais e locais. Além disso, também estão presentes comparações entre o GP-GlobalLocal contra o GP-Global e o GP-Local.

Os resultados mostram que o GP-Global possui resultados similares ao melhor descritor global, mas é sempre melhor que o local. Pode-se observar também que a efetividade dos resultados do GP-Local é limitada pelos resultados ruins de alguns descritores locais, razão pela qual a combinação destes não apresenta melhoras significativas. Apesar do baixo desempenho dos descritores locais, a sua combinação com descritores globais supera todos os outros resultados.

Na Figura 4.10, um exemplo de consulta na classe *Starfish* da base Caltech25 é apresentado, com acertos representados por retângulos em volta das imagens. Neste

Tabela 4.13: Diferença entre o MAP das diferentes abordagens com uma confiança de 95%.

Abordagem	*FreeFoto*		*Caltech*		*Youtube*	
	min.	max.	min.	ma x.	min.	max.
GP-Global – Melhor Global	-0,0201	0,1031	0,0091	0,0333	-0,0031	0,0168
GP-Global – Melhor Local	0,0617	0,1904	0,0049	0,0911	0,0067	0,0630
GP-Local – Melhor Global	-0,0503	0,0238	-0,0345	0.0103	-0,0603	0,0018
GP-Local – Melhor Local	0,0377	0,0273	-0,0102	0,0396	-0,0043	0,0019
GP-GlobalLocal – Melhor Global	-0,0229	0,1355	0,0131	0,0633	0,0052	0,0213
GP-GLobalLocal – Melhor Local	0,0887	0,1929	0,0139	0,1161	0,0148	0,0678
GP-GlobalLocal – GP-Global	-0,0078	0,0375	0,0024	0,0315	0,0001	0,0128
GP-GlobalLocal – GP-Local	0,0728	0,1881	0,0184	0,0822	0,0151	0,0699

Tabela 4.14: Diferença entre o P@5 das diferentes abordagens com uma confiança de 95%.

Approach	*FreeFoto*		*Caltech*		*Youtube*	
	min.	max.	min.	max.	min.	max.
GP-Global – Melhor Global	0,0139	0,0709	-0,0050	0,0529	-0,0040	0,0218
GP-Global – Melhor Local	0,0967	0,2402	0,0126	0,1650	0,0183	0,0777
GP-Local – Melhor Global	-0,1868	-0,0352	-0,0651	0,0100	-0,0721	-0,0101
GP-Local – Melhor Local	-0,0047	0,0348	-0,0214	0,0961	-0,0060	0,0060
GP-GlobalLocal – Melhor Global	0,0197	0,0910	0,0197	0,0707	0,0055	0,0362
GP-GLobalLocal – Melhor Local	0,1214	0,2415	0,0301	0,1900	0,0344	0,0855
GP-GlobalLocal – GP-Global	-0,0054	0,0313	0,0044	0,0380	-0,0002	0,0241
GP-GlobalLocal – GP-Local	0,1121	0,2207	0,0347	0,1107	0,0374	0,0865

exemplo, o melhor descritor global e o GP-Global tiveram uma resposta correta, além da consulta, ambos na segunda posição. O melhor descritor local e o GP-Local não tiveram acertos, além da consulta. O GP-GlobalLocal teve duas respostas corretas, além da consulta, nas segunda e terceira posições. Este exemplo indica que o GP-GlobalLocal utilizou a mesma informação que o GP-Global, como pode ser visto pela mesma imagem retornada na segunda posição e, além disso, melhorou os resultados. Para esta consulta, o indivíduo $sqrt((sqrt((((0.15*((surf+surfBOF)+siftBOF)))*las)))*((gch*(((((bic+surf)*(las+(bic+siftBOF)))+(sqrt(((0.21*surf)))))*(sqrt((sqrt(las)))))*(siftBOF*siftBOF))) + (((sqrt(las)) + (sqrt(jac))) * (siftBOF * (sqrt(sift)))))))$ foi gerado para o GP-GlobalLocal.

A ideia de que o GP-GlobalLocal utiliza a informação predominante, aquela que é mais relevante para a consulta em questão, também é ilustrada na Figura 4.11, em que o melhor descritor global teve todas as respostas erradas, enquanto o melhor descritor local acertou a segunda e a quarta respostas, entretanto, o GP-Local acertou as três primeiras

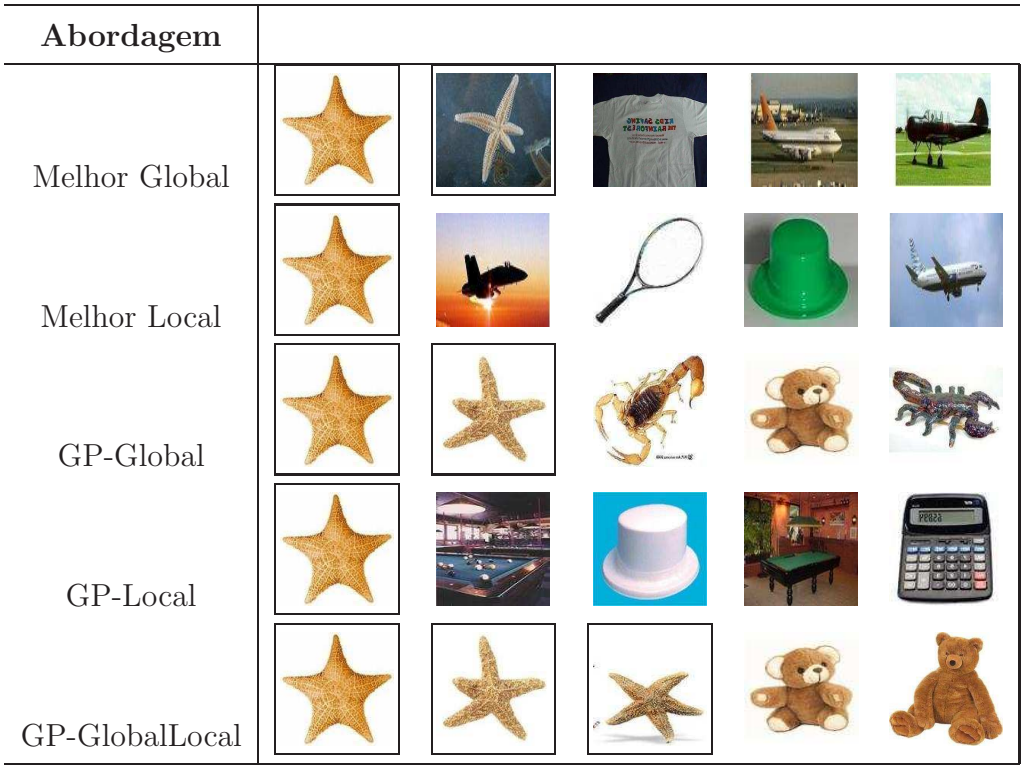


Figura 4.10: Cinco melhores resultados de uma consulta na classe *Starfish* da Caltech.

além da consulta.

Provavelmente, nesta classe, a informação local tem uma resposta melhor, porque globalmente a informação visual confunde-se com as imagens da classe *clouds*, uma vez que as duas classes possuem imagens com o céu azul no fundo, como pode ser visto nas respostas erradas do melhor descritor global e do GP-Global.

Um fato interessante é que uma melhora pode ser observada nas respostas do GP-GlobalLocal em comparação ao GP-Local, possivelmente utilizando informação local, que possui vantagem nesta classe, e informação global, como pode ser visto pelas imagens que aparecem na quarta posição no resultado do GP-Global e em terceiro no resultado do GP-GlobalLocal, mas não aparece em nenhum dos resultados baseados em informação local. Nesta consulta, o indivíduo que gerou estes resultados para o GP-GLobalLocal foi $(sqrt((sqrt(((sqrt(siftBOF)) * jac))))) * ((0.69 * ((sqrt(((sqrt(sift)) + ((0.74 * siftBOF))))) + ((qcch * ((sqrt(surfBOF)) * (sqrt(bic)))) + surf))))$.

Finalmente, no exemplo da Figura 4.12, há um indício de que a combinação dos dois tipos de características pode levar a melhores resultados, apesar do fato de que, isoladamente ambas apresentam resultados insatisfatórios.

Pode-se observar que os descritores e combinações baseados em informação global se confundem com a informação de cor, enquanto os baseados em características locais

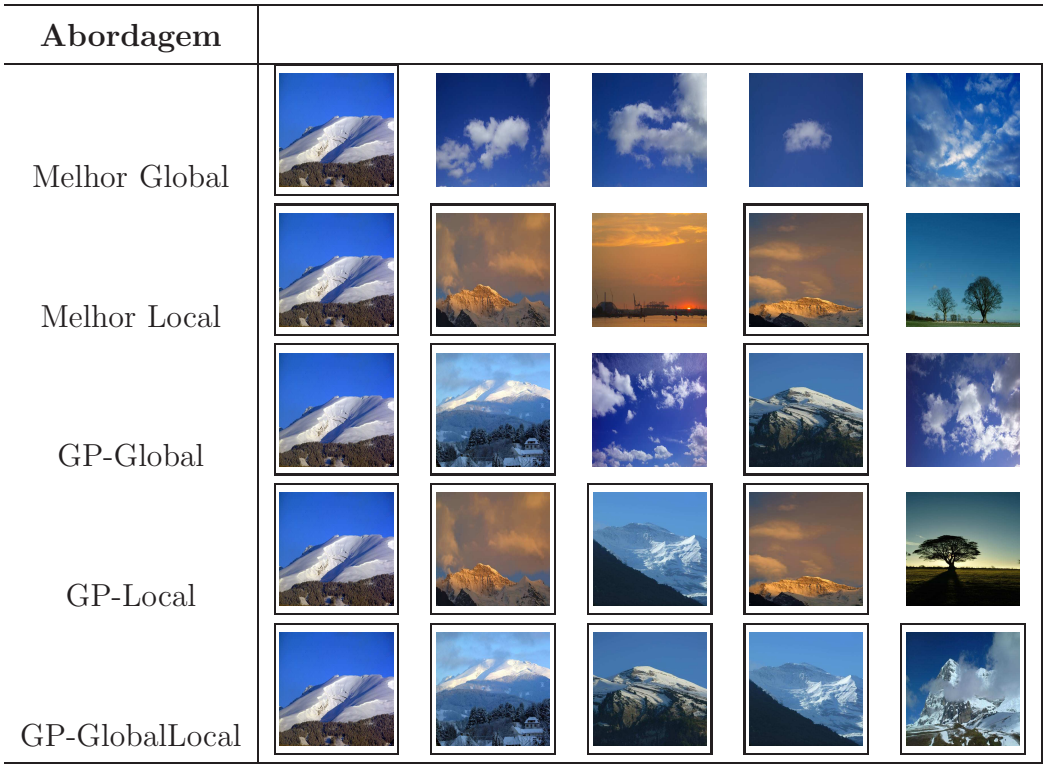


Figura 4.11: Cinco melhores resultados de uma consulta na classe *Mountains* da FreeFoto.

confundem a classe *Leaves* com as classes *Clouds* e *Mountains*. Ambos têm todas as respostas erradas. Ainda assim, quando ambas as características são combinadas, nesta consulta, a segunda e terceira resposta estão corretas. O indivíduo que gerou estes resultados para o GP-GlobalLocal foi $(((sqrt(las)) + (sqrt(((0.31 * jac))))) + (sqrt((jac + (((sqrt(bic)) + ((((sqrt(siftBOF)) + ((bic + surf) + (sqrt((jac + sift)))))) + (sqrt((gch + sift)))) + sift)) + (jac + surf)))))) + siftBOF$.

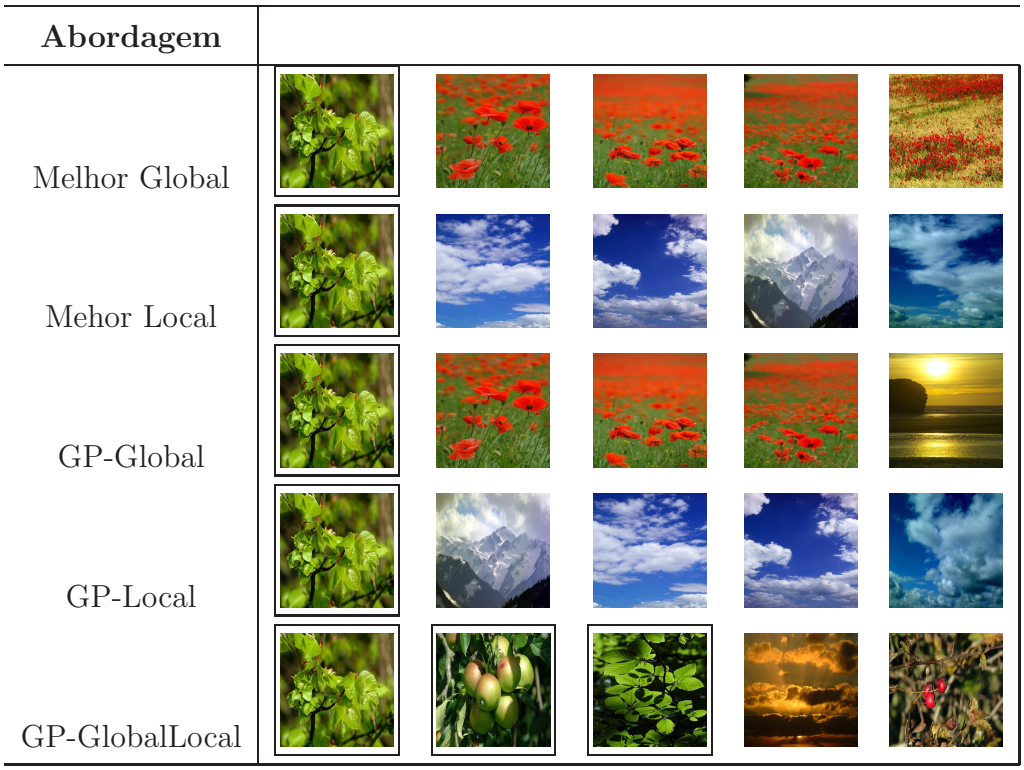


Figura 4.12: Cinco melhores resultados de uma consulta na classe *Leaves* da FreeFoto.

Capítulo 5

Conclusões e Trabalhos Futuros

Este trabalho investigou e analisou o impacto da combinação de diferentes tipos de descritores para recuperação de imagens e vídeos. Um arcabouço de programação genética foi utilizado para a combinação de descritores globais e locais. Doze descritores e suas possíveis combinações tiveram uma avaliação de desempenho, cobrindo uma variedade de características visuais.

Os resultados da comparação em três grandes coleções de dados mostraram que características globais e locais possuem informações diferentes e complementares que podem ser exploradas de forma a melhorar os resultados em recuperação. Este trabalho resultou na publicação de um artigo no *"17th Iberoamerican Congress on Pattern Recognition"*, no ano de 2012, intitulado *Fusion of Local and Global Descriptors for Content-Based Image and Video Retrieval*.

Como trabalho futuros, podem-se citar as seguintes tarefas:

- investigar outras formas de caracterizar propriedades visuais para recuperação de imagens e vídeos. Exemplos incluem descritores de forma [1] e relacionamento espacial para imagens [35], ou técnicas de padrões de movimento [32] e descritores baseados em conceitos de vídeos [43].

- como ressaltado na Seção 2.2.2, a informação espaço-temporal presente nos vídeos é de muita importância. Dessa forma, a introdução deste tipo de característica para combinação com as outras poderia ser benéfica. Um exemplo de descritor com essas características é o STIP, apresentado na Seção 2.3.3.

- o arcabouço de programação genética apresenta diversos parâmetros que influenciam o resultado final da combinação. A exploração destes parâmetros deve ser realizada para encontrar a configuração mais adequada possível de forma a obter os melhores resultados. Como o número de parâmetros a ser escolhido é muito grande, uma

forma de realizar este estudo é a fixação da maioria dos parâmetros e a variação apenas dos parâmetros relacionados a uma mesma função como, por exemplo, o número de indivíduos.

- o arcabouço GP utilizado separa a base em duas partes como conjunto de treinamento e conjunto de teste. No entanto, este tipo de configuração pode enfrentar um problema chamado de *overfitting*. Para tratar tal tipo de problema, o uso do arcabouço GP poderia ser considerado com a base separada em três partes, treinamento, validação e teste, como apresentado no Algoritmo 2.

- o uso de histogramas de palavras visuais também apresenta alguns parâmetros que podem ser variados. Um estudo da construção de dicionários de palavras visuais com diferentes configurações poderia ser realizado.

- diversas outras aplicações poderiam se beneficiar da combinação de descritores locais e globais. Em geral, o uso da estratégia de combinação adotada aqui em outras aplicações dependerá da definição de uma função de *fitness* apropriada. Exemplos de possíveis aplicações incluem classificação de imagens, detecção de objetos em vídeo e a determinação da localização geoespacial em vídeos.

- o GP é uma das técnicas de combinação de características existentes. Outros métodos de combinação poderiam ser utilizados para agregar a informação das características globais e locais como, por exemplo, *Support Vector Machines* [4] e regras de associação [2].

Bibliografia

[1] N. Arica and F. T. Y. Vural. BAS: A Perceptual Shape Descriptor Based on the Beam Angle Statistics. *Pattern Recognition Letters*, 24(9-10):1627–1639, June 2003.

[2] G. M. Armigliatto. Anotação Automática de Imagens Utilizando Regras de Associação. Master's thesis, Instituto de Computação, Universidade Estadual de Campinas, 2011.

[3] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded Up Robust Features. In A. Leonardis, H. Bischof, and A. Pinz, editors, *Computer Vision – ECCV 2006*, volume 3951 of *Lecture Notes in Computer Science*, pages 404–417. Springer Berlin / Heidelberg, 2006.

[4] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A Training Algorithm for Optimal Margin Classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pages 144–152, New York, NY, USA, 1992. ACM.

[5] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning Mid-Level Features for Recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2559–2566, 2010.

[6] S. M. P. Caceres. Técnicas de Visualização para Sistemas de Recuperação de Imagens por Conteúdo. Master's thesis, Instituto de Computação, Universidade Estadual de Campinas, 2010.

[7] R. da S. Torres and A. X. Falcão. Content-Based Image Retrieval: Theory and Applications. *Revista de Informática Teórica e Aplicada*, 13(2):161–185, 2006.

[8] R. da S. Torres, A. X. Falcão, M. A. Gonçalves, J. P. Papa, B. Zhang, W. Fan, and E. A. Fox. A Genetic Programming Framework for Content-based Image Retrieval. *Pattern Recognition*, 42(2):283 – 292, 2009. Learning Semantics from Multimedia Content.

[9] J. dos Santos, C. Ferreira, R. da S. Torres, M. Gonçalves, and R. Lamparelli. A Relevance Feedback Method based on Genetic Programming for Classification of Remote Sensing Images. *Information Sciences*, 181(13):2671 – 2684, 2011. Including Special Section on Databases and Software Engineering.

[10] M. d'Aquin and N. F. Noy. Where to publish and find ontologies? a survey of ontology libraries. *Web Semantics: Science, Services and Agents on the World Wide Web*, 11(0):96 – 111, 2012.

[11] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88:303–338, 2010. 10.1007/s11263-009-0275-4.

[12] W. Fan, E. A. Fox, P. Pathak, and H. Wu. The Effects of Fitness Functions on Genetic Programming-Based Ranking Discovery for Web Search. *Journal of the American Society for Information Science and Technology*, 55(7):2004, 2004.

[13] F. A. Faria. Uso de Técnicas de Aprendizagem para Classificação e Recuperação de Imagens. Master's thesis, Instituto de Computação, Universidade Estadual de Campinas, 2010.

[14] C. Ferreira, J. Santos, R. da S. Torres, M. Gonçalves, R. Rezende, and W. Fan. Relevance Feedback based on Genetic Programming for Image Retrieval. *Pattern Recognition Letters*, 32(1):27 – 37, 2011. Image Processing, Computer Vision and Pattern Recognition in Latin America.

[15] C. D. Ferreira. Recuperação de Imagens com Realimentação de Relevância Baseada em Programação Genética. Master's thesis, Instituto de Computação, Universidade Estadual de Campinas, 2007.

[16] X. Gao, X. Li, J. Feng, and D. Tao. Shot-based Video Retrieval with Optical Flow Tensor and HMMs. *Pattern Recognition Letters*, 30(2):140 – 147, 2009. Video-based Object and Event Analysis.

[17] P. Geetha and V. Narayanan. A Survey of Content-Based Video Retrieval. *Journal of Computer Science*, 4(6):474–486, 2008.

[18] M. Helala, M. Selim, and H. Zayed. An Image Retrieval Approach Based on Composite Features and Graph Matching. In *Second International Conference on Computer and Electrical Engineering*, volume 1, pages 466 –473, dec. 2009.

[19] C.-B. Huang and Q. Liu. An Orientation Independent Texture Descriptor for Image Retrieval. In *International Conference on Communications, Circuits and Systems*, pages 772–776, july 2007.

[20] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih. Image Indexing Using Color Correlograms. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 762–, Washington, DC, USA, 1997. IEEE Computer Society.

[21] P. Indyk and N. Thaper. Fast Image Retrieval via Embeddings. In *3rd International Workshop on Statistical and Computational Theories of Vision*. ICCV, 2003.

[22] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA, 1992.

[23] M. Kunt. Special Issue on: Content-based Visual Information Retrieval. *Signal Processing*, 82(2):327 – 327, 2002.

[24] I. Laptev. On Space-Time Interest Points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005.

[25] D. Liu and T. Chen. Video Retrieval based on Object Discovery. *Computer Vision and Image Understanding*, 113(3):397 – 404, 2009. ¡ce:title¿Special Issue on Video Analysis¡/ce:title¿.

[26] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma. A Survey of Content-based Image Retrieval with High-Level Semantics. *Pattern Recognition*, 40(1):262 – 282, 2007.

[27] D. G. Lowe. Object Recognition from Local Scale-Invariant Features. *IEEE International Conference on Computer Vision*, 2:1150, 1999.

[28] C. D. Manning, P. Raghavan, and H. Schtze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[29] B. M. Mehtre, M. S. Kankanhalli, A. D. Narasimhalu, and G. C. Man. Color Matching for Image Retrieval. *Pattern Recognition Letters*, 16(3):325 – 331, 1995.

[30] K. Mikolajczyk and C. Schmid. A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615 – 1630, 2005.

[31] C. Morand, J. Benois-Pineau, and J.-P. Domenger. HD Motion Estimation in a Wavelet Pyramid in JPEG2000 Context. In *15th IEEE International Conference on Image Processing*, pages 61 –64, oct. 2008.

[32] C. Morand, J. Benois-Pineau, J.-P. Domenger, J. Zepeda, E. Kijak, and C. Guillemot. Scalable Object-based Video Retrieval in HD Video Databases. *Signal Processing: Image Communication*, 25(6):450 – 465, 2010.

[33] H. Pedrini and W. Schwartz. *Análise de Imagens Digitais: Princípios, Algoritmos e Aplicações*. Editora Thomson Learning, 2007.

[34] O. A. B. Penatti. Estudo Comparativo de Descritores para Recuperação de Imagens por Conteúdo na Web. Master's thesis, Instituto de Computação, Universidade Estadual de Campinas, 2009.

[35] O. A. B. Penatti, E. Valle, and R. da S. Torres. Encoding Spatial Arrangement of Visual Words. In *16th Iberoamerican Congress conference on Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 240–247, Berlin, Heidelberg, 2011. Springer-Verlag.

[36] M. Petkovic. Content-based Video Retrieval. In *VII Conference on Extending Database Technology*, 2000.

[37] W. Ren, S. Singh, M. Singh, and Y. Zhu. State-of-the-Art on Spatio-Temporal Information-based Video Retrieval. *Pattern Recognition*, 42(2):267 – 282, 2009. Learning Semantics from Multimedia Content.

[38] Y. Rubner, C. Tomasi, L. J. Guibas, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[39] A. Salgian. Combining Local Descriptors for 3D Object Recognition and Categorization. *19th International Conference on Pattern Recognition*, pages 1 – 4, 2008.

[40] N. C. Simões. Detecção de Algumas Transições Abruptas em Sequências de Imagens. Master's thesis, Instituto de Computação, Universidade Estadual de Campinas, 2004.

[41] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-Based Image Retrieval at the End of the Years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, December 2000.

[42] R. O. Stehling, M. A. Nascimento, and A. X. Falcão. A Compact and Efficient Image Retrieval Approach based on Border/Interior Pixel Classification. In *Eleventh International Conference on Information and Knowledge Management*, pages 102–109, New York, NY, USA, 2002. ACM.

[43] J.-H. Su, Y.-T. Huang, H.-H. Yeh, and V. S. Tseng. Effective COntent-based Video Retrieval using Pattern-Indexing and Matching Techniques. *Expert Systems with Applications*, 37(7):5068 – 5085, 2010.

[44] M. Swain and D. Ballard. Indexing Via Color Histograms. In *Third International Conference on Computer Vision*, pages 390 –393, dec 1990.

[45] B. Tao and B. W. Dickinson. Texture Recognition and Image Retrieval Using Gradient Indexing. *Journal of Visual Communication and Image Representation*, 11(3):327 – 342, 2000.

[46] E. Valle and M. Cord. Advanced Techniques in CBIR: Local Descriptors, Visual Dictionaries and Bags of Features. *Tutorials of the XXII Brazilian Symposium on Computer Graphics and Image Processing*, pages 72–78, 2009.

[47] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J. M. Geusebroek. Visual Word Ambiguity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1271–1283, 2010.

[48] F. Wang. A survey on automatic image annotation and trends of the new age. *Procedia Engineering*, 23(0):434 – 438, 2011. ¡ce:title¿PEEA 2011¡/ce:title¿.

[49] H. Wang, M. M. Ullah, A. Klaser, I. Laptev, and C. Schmid. Evaluation of Local Spatio-Temporal Features for Action Recognition. In *British Machine Vision Conference*, London, Royaume-Uni, Sept. 2009. CLASS.

[50] R. Weber, H. Schek, and S. Blott. A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces. In *24rd International Conference on Very Large Data Bases*, pages 194–205, 1998.

[51] A. Williams and P. Yoon. Content-based Image Retrieval using Joint Correlograms. *Multimedia Tools and Applications*, 34(2):239–248, 2007.

[52] Y. Wu and Y. Wu. Shape-Based Image Retrieval Using Combining Global and Local Shape Features. *2nd International Congress on Image and Signal Processing*, pages 1 – 5, 2009.

[53] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo. Evaluating Bag-of-Visual-Words Representations in Scene Classification. In *International Workshop on Multimedia Information Retrieval*, pages 197–206, New York, NY, USA, 2007. ACM.

[54] Y. Zhai, J. Liu, X. Cao, A. Basharat, A. Hakeem, S. Ali, M. Shah, C. Grana, and R. Cucchiara. Video Understanding and Content-Based Retrieval. In *TREC Video Retrieval Evaluation Workshop Online Proceedings*, 2005.

[55] M. Zhu. Recall, Precision and Average Precision. Technical report, Working Paper 2004-09, University of Waterloo, 2004.