
**UNIVERSIDADE SALVADOR – UNIFACS**
**PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO**
**MESTRADO ACADÊMICO EM SISTEMAS E COMPUTAÇÃO**

**CARLOS ALBERTO FRAGA PIMENTEL FILHO**

# UM AMBIENTE PARA INDEXAÇÃO E RECUPERAÇÃO DE CONTEÚDO DE VÍDEO BASEADO EM CARACTERÍSTICAS VISUAIS

Salvador
2008

**CARLOS ALBERTO FRAGA PIMENTEL FILHO**

# UM AMBIENTE PARA INDEXAÇÃO E RECUPERAÇÃO DE CONTEÚDO DE VÍDEO BASEADO EM CARACTERÍSTICAS VISUAIS

Dissertação apresentada ao Mestrado Acadêmico em Sistemas e Computação da Universidade Salvador – UNIFACS, como requisito parcial para obtenção do grau de Mestre

Orientador: Prof. Dr. Celso Alberto Saibel Santos
Co-Orientador: Prof. Dr. Thomas Araújo Buck

Salvador
2008

FICHA CATALOGRÁFICA
(Elaborada pelo Sistema de Bibliotecas da Universidade Salvador - UNIFACS)

Pimentel Filho, Carlos Alberto Fraga
Um ambiente para indexação e recuperação de conteúdo de vídeo baseado em características visuais/Carlos Alberto Fraga Pimentel Filho - 2008.
124 f.

Dissertação (Mestrado) - Universidade Salvador – UNIFACS. Mestrado em Sistemas de Computação, 2008.
Orientador: Prof. Celso Alberto Saibel Santos

1. Recuperação da informação 2. Indexação de vídeos I. Santos, Celso Alberto Saibel, orient. II. Título.

CDD: 025.524

CARLOS ALBERTO FRAGA PIMENTEL FILHO

# UM AMBIENTE PARA INDEXAÇÃO E RECUPERAÇÃO DE CONTEÚDO DE VÍDEO BASEADO EM CARACTERÍSTICAS VISUAIS

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre em Sistemas e Computação, Universidade Salvador - UNIFACS, pela seguinte banca examinadora:

Celso Alberto Saibel Santos (orientador)
Doutor em Infomatique Fondamentalle et Parallelisme pela Université Paul Sabatier de Toulouse, França
Universidade Salvador – UNIFACS

Jugurta Rosa Montalvão Filho

Doutor em Automatique Et Traitement du Signal pela Universite de Paris XI (Paris-Sud), França

Universidade Federal de Sergipe – UFS

Antonio Lopes Apolinário Junior

Doutor em Engenharia de Sistemas e computação pela Universidade Federal do Rio de Janeiro - UFRJ, Brasil

Universidade Estadual de Feira de Santana – UEFS

Thomas Araújo Buck

Doutem em informática pela Universität Tübingen, Alemanha

Universidade Salvador - UNIFACS

Salvador, 08 de setembro de 2008

Para meus pais Carlos Pimentel e Fátima Maynart e meus tios Evilácio Viana e Sônia Maynart: pessoas que tanto amo, sempre me incentivaram e acreditaram em mim.

Carlos A. Fraga Pimentel Filho

## AGRADECIMENTOS

Primeiramente agradeço a Deus por ter aberto tantas portas pelas quais passei e às vezes nem me dei conta, em cada momento de reflexão posso perceber que o caminho das pedras fora mostrado e quanta luz sempre houve por lá.

Agradeço os meus pais por me incentivarem sempre na busca do conhecimento, por me apoiarem nos maiores desafios e pelo esforço empenhado na construção da minha educação e realização pessoal.

Aos meus tios Evilácio Viana e Sônia Maynart, agradeço não só pelo acolhimento em seu lar durante o período de mestrado, mas também pelos ensinamentos e pelo amor que me deram, vocês foram como pais para mim durante esse período. Sem o apóio dos senhores esse trabalho não teria sido realizado.

Agradeço ao primo Denison Maynart por me emprestar seu espaço e pela serenidade da sua pessoa que sempre esteve disposta a me ouvir, tenho você como meu irmão mais novo.

Ao meu tio Jerônimo Maynart obrigado pelo incentivo e companheirismo, tenho você como meu irmão mais velho, sempre disposto a ouvir e ajudar.

Agradeço ao orientador Celso Saibel pela confiança que me foi depositada ao me aceitar como orientando e pesquisador em seu projeto. Durante esse tempo de convivência aprendi não somente sobre multimídia e tecnologia, mas também sobre a vida acadêmica e a escrita científica. Por fim espero que o sr. veja apenas o lado positivo de nossas divergências.

Ao professor e co-orientador Thomas Buck, obrigado pelas aulas do mestrado e pela co-orientação do trabalho. Especialmente no início das pesquisas suas dicas contribuíram significativamente nos rumos e na realização deste trabalho.

Obrigado ao professor da graduação Almerindo Rehem por me incentivar no ingresso do mestrado e ao professor Jugurta Montalvão desde a iniciação científica, considero que minha formação de mestre começou mesmo na graduação com sua orientação nos trabalhos de IC e monografia, agradeço também pela participação na banca desse trabalho.

Aos amigos do mestrado, agradeço pelo apóio, companhia e bons momentos. Dentre eles, os amigos da terra de Aracaju Fábio e George, os amigos Jeane,

Zanza, Alberto, Alexandre, André, Daniel e todos os outros amigos da equipe NUPERC em especial Danila, que se tornou uma pessoa importante para mim.

Agradeço aos colegas e amigos: Welington pela parceria nesse trabalho de pesquisa e ajuda nas avaliações de resultados; e ao amigo Alex Bennes pelas dicas e implementações em C++ no uso da API do DirectX.

Obrigado aos demais professores do mestrado e aos colaboradores, em especial, Jane e Isabel, tanto no apóio ao bom andamento dos trabalhos e estudos quanto na companhia durante os intervalos de almoço.

E por fim, agradeço a instituição governamental do CAPES pelo financiamento integral dessa pós-graduação.

## RESUMO

O presente trabalho está centrado na área de recuperação informação em vídeos com base em suas características visuais. Ele propõe um ambiente para sumarização e indexação automática de vídeos digitais para dar suporte a operações de busca baseada no conteúdo visual em um repositório de vídeos. A técnica de sumarização automática é baseada numa avaliação do ponto de equilíbrio entre perda de quadros e redundância, considerando os aspectos visuais dos quadros. Como resultado, é possível se obter uma representação bastante reduzida do vídeo através de quadros-chave, que armazenam informações suficientes sobre as características visuais do conteúdo do vídeo. A partir desse conjunto reduzido de quadros-chave, são extraídas medidas estatísticas e uma entidade chamada de *assinatura da imagem* para caracterizar o conteúdo visual dos vídeos. A assinatura da imagem tem a capacidade de representar cada quadro-chave de modo ainda mais compacto, ao mesmo tempo em que mantém os aspectos visuais que o caracterizam, provendo um recurso poderoso para a indexação. A abordagem descrita foi aplicada a um repositório contendo mais de 34 horas de vídeos jornalísticos, representados de forma reduzida por cerca de 1,25% do número total de quadros. Essa representação reduzida foi então utilizada para indexar o conteúdo dos vídeos, servindo de base para diversos experimentos, bem como para análise do desempenho da recuperação de conteúdo visual em vídeos com a aplicação das diversas técnicas implementadas no ambiente.



Palavras-chave: Vídeo digital. Indexação de vídeo. Sumarização de vídeo. CBIR. CVIR.

**ABSTRACT**

This work is centered in the area of video information retrieval based on its visual characteristics. It proposes an environment for automatic summarization and indexing of digital videos in order to support the operations of query based on visual content in a video repository. The technique of automatic summary is based on a benchmark avaliation that consider balance between key frame losing and redundance based on visual aspects. As a result, it is possible to obtain a very low representation of the video through key frames, which store enough information about the visual characteristics of the video content. From this small set of key frames, statistics measures and an entity called image signature are taken to characterize the visual content of the videos. The image signature has the ability to represent each key frame in a even more compact way, while maintaining the visual aspects that characterize it, providing a powerful tool for indexation. The approach described was applied to a repository containing more than 34 hours of journalistic videos, thus compressed by about 1.25% of the total number of frames. This reduced representation was then used to index the video contents, providing the basis for various experiments and analysis of performance for the recovery of visual content in videos with the application of various techniques implemented in this work.

Keywords: Digital video. Video indexing. Video summarization. CBIR. CVIR.

## LISTA DE FIGURAS

## LISTA DE TABELAS

## LISTA DE GRÁFICOS

## SUMÁRIO

# 1 INTRODUÇÃO

## 1.1MOTIVAÇÃO

A recuperação de informação, como uma área de estudo, já existe há algum tempo, contudo, o foco inicial durante um bom período havia sido dado ao desenvolvimento de sistemas de recuperação de informação textual. Nesse caso, as principais questões a serem respondidas eram: como extrair palavras chave de um documento? Como categorizá-lo, como resumi-lo ou recuperá-lo? Enfim, todo estudo ligado à recuperação de informação esteve inicialmente direcionado ao tratamento de documentos textuais (WANG; LIU; HUANG, 2000).

Em 1951, o pesquisador Moores Calvin (1951), cunhou o termo "*Information Retrieval*" ou "Recuperação de Informação". Este termo foi proposto para descrever o processo sobre o qual, um conjunto de definições pode converter uma requisição de informação num conjunto de referências de fato úteis. Moores descreve a "Recuperação de Informação" em algo como: "abraçar os aspectos intelectuais da descrição da informação e suas especificações para busca, e também qualquer tipo de sistema, técnica, ou máquinas empregadas em permitir esse tipo de operação". Muito embora Moores tenha se referido à busca voltada ao documento textual, sua definição estende-se perfeitamente à busca em outros tipos de mídia (GUPTA; JAIN, 2007).

Dentre todos os tipos de mídia (texto, imagem, gráfico, áudio e vídeo), o vídeo é a mídia mais desafiadora para os pesquisadores da área de recuperação de informação porque pode combinar todas as outras mídias num único fluxo de dados (TIAN, 2006). O vídeo também é a mídia mais efetiva para captura do mundo ao nosso redor. Combinando áudio, imagem e efeitos visuais, o vídeo pode representar um alto grau de realidade (GUAN; KUNG; LARSEN, 2000). A mídia áudio-visual também é importante no entretenimento e na disseminação da informação. Além do entretenimento, os vídeos auxiliam e enriquecem diversas áreas como educação (tradicional e à distância), esportes, medicina e outras.

A motivação para o presente trabalho vem do projeto de indexação e recuperação de conteúdo chamado de Ambiente para Descrição, Indexação e

Consulta de Conteúdos de VÍdeos Digitais (DEIVID), desenvolvido desde Fevereiro de 2006 na Universidade Salvador (UNIFACS). Com apoio do CNPq[1], o projeto trata de um tema bastante atual na área da computação: o desenvolvimento de sistemas para gerenciamento de grandes volumes de dados multimídia, em especial vídeos. Recentemente, a Sociedade Brasileira de Computação através de um relatório, apontou que um dos 5 maiores desafios para a área de computação nos próximos anos é a Gestão da Informação em grandes volumes de dados multimídia distribuídos (SBC, 2007). Isto porque o imenso conjunto heterogêneo de dados multimídia produzido pela sociedade atual precisa ser processado, armazenado e disponibilizado para tornar possível a extração de informação para os mais diferentes tipos de usuários. Vencer esse desafio exige a aplicação de novas técnicas e métodos de gerenciamento, extração de semântica de conteúdos audiovisuais, além da integração, indexação e recuperação de dados e informação, com o uso de soluções escaláveis.

Neste contexto, o projeto DEIVID tem como focos principais: a modelagem, a implementação e a avaliação de um ambiente voltado ao gerenciamento, busca e recuperação do conteúdo de vídeos digitais. A ambição do projeto é identificar e propor mecanismos que facilitem a exploração efetiva (gerenciamento, busca e recuperação) dos vídeos digitais através do acesso orientado ao conteúdo desses vídeos. Assim, o projeto se propôs também a estudar formas eficientes de utilização das descrições obtidas para facilitar (i) a interação dos usuários (consulta, edição, visualização) com estes conteúdos, (ii) a gerência de bases de conteúdos multimídia e também, (iii) a geração de apresentações personalizadas dos vídeos manipulados, ou apresentações multimídia com a semântica aumentada através das anotações associadas a partes do vídeo.

## 1.2 OBJETIVOS

Dentre os objetivos inicialmente propostos dentro do projeto DEIVID, essa dissertação está centrada num subconjunto que envolve: (i) a interação dos usuários (consulta e visualização) com um repositório de vídeos; (ii) a extração e indexação automática de características visuais dos conteúdos dos vídeos e (iii) o acesso a tal conteúdo de modo eficiente e rápido.

---

[1] Projeto Nº 506647/2004-8 aprovado no Edital CT-INFO: CNPq 31/2004 – PDPG-TI, iniciado em 2006.

A ambição principal do presente trabalho é criar um ambiente onde as pessoas possam extrair, sumarizar, armazenar, navegar e recuperar conteúdo visual em bases de vídeos. Seu foco principal é a busca de quadros-chave de vídeo, tentando tornar o acesso ao conteúdo visual tão fácil quanto se pode recuperar dados em texto (GUPTA; JAIN, 2007). A avaliação da solução implementada através de experimentos foi realizada sobre um conjunto contendo cerca de 34 horas de vídeos jornalísticos, capturados a partir das transmissões abertas do Jornal Nacional da Rede Globo.

Como objetivos específicos, o presente trabalho se propôs a:

a) criar uma ferramenta de *parsing* onde são extraídas medidas estatísticas, sumários e representações compactas do vídeo para análise posterior. A ferramenta de *parsing* deve gerar informação suficiente para a inferência de segmentação temporal do vídeo como cortes de tomada;
b) geração de uma representação compacta e sumarizada do vídeo com quadros-chave, os quais são por sua vez transformados em metadados que o representam;
c) inserir os metadados obtidos com o processo de *parsing* num repositório de dados (um banco de dados relacional) que possa indexar o conteúdo e recuperá-lo de forma rápida e precisa;
d) criar uma ferramenta que permita analisar automaticamente os dados gerados pelo *parsing* e por sua vez gerar a indexação do vídeo, segmentando-o na dimensão temporal;
e) e por último, prover uma interface gráfica onde o usuário possa interagir com a ferramenta efetuando busca de conteúdo visual e navegação nos vídeos indexados.

## 1.3 JUSTIFICATIVA

O avanço das tecnologias de produção, transmissão, distribuição e armazenamento de vídeos resultou na criação de grandes repositórios, suscitando, porém, diversas questões importantes. Como lidar, navegar e recuperar informação de forma rápida e eficiente nesse tipo de mídia? Como buscar uma cena ou um quadro específico dentro do conteúdo do vídeo? Com questões como essas, a

análise automática de conteúdo multimídia vem se tornando uma área de pesquisa cada vez mais importante e explorada (SEINSTRA *et al*., 2007).

Além das questões anteriores, outras ainda continuam em aberto. É possível classificar gêneros e subgêneros proveniente de imagens a partir de uma análise estatística? Há possibilidade de se encontrar automaticamente objetos em vídeos a partir da análise estatística em uma grande escala de volume de dados (SEINSTRA *et al*., 2007)? A solução para essas e outras questões podem ser potencialmente aplicadas numa vasta gama de áreas como: recuperação de informação, biomedicina, comércio, educação, bibliotecas digitais e busca na *world wide web* (LI, 2003). Por exemplo, o *Netherlands Forensic Institute* tem uma necessidade imperativa de realizar a detecção de objetos e de indivíduos em vídeos obtidos por câmeras de vigilância eletrônica (SEINSTRA *et al*., 2007).

A vantagem da tecnologia digital é que a mesma torna possível o uso de novas estruturas de navegação e busca de conteúdo em vídeos. Estas tecnologias são potencialmente, e na prática, mais avançadas que aquelas conhecidas nos antigos sistemas de fitas de vídeo. Nesses sistemas, a busca de conteúdo ocorre tipicamente de modo seqüencial com base na inspeção visual do vídeo. Por outro lado, o formato digital pode agregar outras técnicas de busca e navegação em vídeo melhorando o gerenciamento desses repositórios como é discutido no decorrer desse trabalho.

Assim, os sistemas multimídia digitais são cada vez mais importantes, pois, as mídias de vídeos e imagens são consideradas “produção de informação de primeira classe”. A premissa básica por trás dos sistemas de recuperação de informação visual sobre essas mídias é dada pelo fato de que os usuários deveriam ser capazes de recuperar seu conteúdo tão facilmente quanto as buscas são feitas em conteúdo textual, e sem a necessidade de anotação manual de vídeo e/ou imagem (GUPTA; JAIN., 2007) e essa é a grande proposta do presente trabalho.

## 1.4 CONTEXTO DO TRABALHO

Ao contrário dos computadores, os seres humanos possuem uma característica cognitiva de poder extrair informação semântica e de relacionar fatos e objetos a partir do mundo visual ao seu redor. As máquinas, entretanto, são melhores que os humanos em mensurar propriedades matemáticas ou estatísticas e retê-las em grandes partições de memória (FLICKNER *et al*., 1997). Essa vantagem

das máquinas sobre os seres humanos é explorada no presente trabalho com a extração de características matemáticas e estatísticas a partir de quadros-chave de vídeos armazenando-as num repositório de dados.

Assim como na recuperação textual o usuário fornece uma chave ou segmento daquilo que pretende recuperar, na recuperação baseada em similaridade visual, o usuário fornece uma imagem similar ao conteúdo que pretende trazer afora. Essa imagem pode ser confeccionada pelo usuário, sob a forma de um rascunho ou pode ser uma imagem exemplo obtida do mundo real. O presente trabalho apresenta um ambiente em que é possível esse tipo de busca visual provendo a recuperação de quadros-chave com base numa imagem de busca previamente fornecida, imagem essa, similar ao conteúdo esperado. Nesse caso, o alvo são os quadros-chave capturados dos vídeos sumarizados e indexados num repositório apropriado.

No caso de uso de recuperação de quadros fornecendo-se uma imagem criada como rascunho, ou simplesmente **busca por rascunho**, subentende-se que o usuário já conheça a base de vídeos e precise acessar uma determinada cena. Entretanto, o mesmo não se lembra exatamente em qual ponto do vídeo esse quadro-chave se encontra. O usuário então desenha um quadro, da melhor maneira como ele recorda a cena, criando um modelo visual do que se encontra em sua memória. O rascunho do quadro desenhado é então passado ao ambiente de busca que por sua vez retorna os quadros-chave dos vídeos que contém conteúdo visual similar ao rascunho.

As propostas apresentadas no presente trabalho podem ser potencialmente aplicadas, por exemplo, na solução de problemas relacionados à busca de conteúdo visual na *world wide web*. A maior parte das soluções atuais para a recuperação de vídeo e imagem na *Web* não examinam o conteúdo em baixo nível, ou seja, os *pixels* que formam as imagens. Em outras palavras, a busca de conteúdo multimídia na *Web* é, ainda hoje, fortemente baseada em metadados, legendas e outras informações textuais que são anexadas de forma manual aos conteúdos de imagens, vídeos e áudios.

Nos sistemas de recuperação de conteúdo visual *Visual Information Retrieval* ou (*VIR*), é importante indexar esse conteúdo a partir de informações que o descrevam visualmente (distribuição de cores, entropia, número e tipo de objetos ou segmentos, etc). Esses sistemas, também chamados de *Content Based Image*

*Retrieval* (*CBIR*) algo como sistemas de “recuperação de imagem com base em conteúdo”. Mais especificamente, no caso do presente trabalho, sistemas do tipo *Content Based Video Retrieval* (CBVR), isto é, sistemas de recuperação de vídeo com base em seu conteúdo. O principal propósito dos sistemas baseados em CBVR é extrair informações que sumarizem o conteúdo do vídeo ao máximo possível, mantendo, contudo, a sua essência (SEINSTRA *et al*., 2007).

Em algumas situações, os resultados obtidos a partir de sistemas CBVR não apenas complementam a busca textual, mas também permitem avanços nas operações de busca que os textos não podem prover. Por exemplo, suponha que uma agência de propaganda tenha um repositório com mais de 50 mil clipes. Um cliente quer encomendar uma nova campanha para comemoração dos 30 anos de sua marca e tem uma exigência: ele deseja uma compilação com “o melhor” dos comerciais antigos da marca, em especial, um trecho de aproximadamente 15 segundos de uma campanha de sucesso, onde aparece um avião, desenha a logomarca da sua companhia com o rastro da fumaça. Embora as pessoas possam formar mentalmente uma idéia da cena, a descrição textual da busca de segmentos ou imagens para o clipe, usando um mecanismo baseado em texto para tal, não é nada trivial. Além disso, o resultado da busca desse clipe, através de palavras chave é obviamente frustrante. A dificuldade nesse caso se deve ao fato de ser impossível garantir que a equipe que anota o vídeo e o usuário possam editar uma busca *ad hoc* de modo a expressar o clipe da mesma maneira (GUPTA; JAIN, 2007). A interpretação da imagem ou da cena, e o nível de detalhes variam até com a mesma pessoa, a depender do tempo passado entre uma visualização e outra, ou da situação e ambiente em que a pessoa se encontra. Sendo assim, espera-se ainda mais divergência entre as interpretações do conteúdo feitas pelo anotador do vídeo e pelo consumidor que realiza as buscas, quando estas operações são realizadas por pessoas distintas. Outro problema é que algumas características visuais são impossíveis de serem descritas textualmente. Por exemplo, como o anotador do vídeo poderia descrever uma cena com 40% de azul, 20% de vermelho, alto contraste e muito brilho? Mas muito brilho o quanto? Esses aspectos são muito subjetivos e difíceis de se descrever textualmente (ZHANG *et al*., 1994).

Apesar das claras vantagens das técnicas de CBVR sobre as baseadas em anotação de conteúdo, as primeiras apresentam algumas desvantagens que não podem ser deixadas de lado. Primeiro, a grande sensibilidade ao ruído. Por exemplo,

pequenas mudanças nos valores dos *pixels* podem levar uma imagem candidata aos primeiros lugares para várias posições abaixo na colocação (*ranking*) da busca. Segundo, a busca é bastante sensível às variações em escala, deslocamento e rotação da imagem. Por exemplo, uma rotação de apenas 15 graus pode impedir que se alcance uma imagem alvo desejada. E terceiro, a variação de iluminação e outros efeitos afetam os *pixels* drasticamente, podendo levar a resultados incorretos na recuperação (GUPTA; JAIN, 2007). Todos esses efeitos, tais como mudança de iluminação, escala (*zoom*), e rotação são inerentemente presentes nos vídeos, fazendo com que um mesmo contexto possa apresentar diferentes versões nessas variações e efeitos.

Por fim, deve-se ressaltar o fato de que a área de recuperação de conteúdo visual é multidisciplinar, uma vez que emprega, simultaneamente, várias áreas da computação, dentre elas: computação visual; processamento de imagem; bancos e estruturas de dados; análise de dados, gerenciamento de informação e sistemas de recuperação (GUPTA; JAIN, 2007). Isso exige um trabalho de integração de várias especialidades para a construção de uma solução efetiva para o problema da recuperação de vídeos baseada em conteúdo.

## 1.5 CONTRIBUIÇÕES

Além da tradicional contribuição decorrente da revisão da literatura nas áreas de recuperação de conteúdo visual e estruturação do conteúdo de vídeos, o trabalho apresenta três outras importantes contribuições relacionadas às etapas de desenvolvimento do ambiente de recuperação proposto.

A primeira dessas contribuições é uma comparação entre dois métodos tradicionais de corte de tomada amplamente descritos na literatura com um método baseado na comparação de quadros utilizando a técnica de assinaturas de *wavelets*, como será detalhado oportunamente na seqüência do texto.

A segunda contribuição está relacionada à captura e à extração de quadros-chave que irão dar suporte a busca de quadros. Essa extração é feita de modo a equilibrar uma relação entre baixa redundância nos quadros capturados e baixa perda de quadros que representem o conteúdo visual do vídeo como um todo. É mostrado nesse trabalho que, a alteração de uma dessas variáveis isoladamente (perda ou redundância) implica automaticamente na “perturbação” da outra. Vários métodos foram confrontados visando a melhor relação de equilíbrio entre essas

variáveis, produzindo assim, uma captura de quadros adequada para os métodos de busca de quadros.

Finalmente, como principal contribuição, esse trabalho compara métodos de busca de quadros de vídeo baseado em diferentes técnicas de extração e comparação de características visuais de imagem descritas na literatura. Embora as técnicas originalmente tenham sido propostas para busca em repositórios de imagens, o trabalho mostra que estas técnicas podem ser também aplicadas com sucesso nas operações de busca orientada ao conteúdo dos vídeos.

## 1.6 ORGANIZAÇÃO DO TRABALHO

O presente trabalho está organizado da seguinte maneira. O capítulo um traz uma introdução com a motivação, os objetivos, a justificativa, o contexto e as contribuições científicas relevantes trazidas pela pesquisa. O capítulo 2 apresenta uma revisão bibliográfica sobre a estruturação de vídeos digitais e principalmente que trabalhos podem ser efetuados em cima do vídeo afim de segmentá-lo, descrevê-lo e representá-lo. O capítulo 3 traz uma introdução na busca de conteúdo em vídeo, com ênfase na busca baseada em conteúdo visual dos quadros do vídeo, também são mostrados os trabalhos relacionados com essa área e formas de comparação de características. No capítulo 4, uma das ferramentas matemáticas mais importantes para a realização dessa pesquisa – a transformada *wavelet* – é apresentada, juntamente com as principais características dessa ferramenta e a sua possível utilização na análise de quadros do vídeo. O final do capítulo apresenta uma métrica de comparação entre quadros de vídeo por um método chamado de assinatura de imagem, que é utilizada para caracterizar os quadros de um vídeo. O capítulo 5 apresenta o ambiente proposto para indexação e recuperação de conteúdo visual de vídeos, as ferramentas implementas e seus recursos. O capítulo 6 mostra o estudo de caso da pesquisa, com os experimentos e resultados das 3 principais contribuições do trabalho. Finalmente, o capítulo 7 encerra a dissertação tecendo as conclusões e apresentando as perspectivas de continuidade dos trabalhos ligados ao tema.

## 1.7 NOTAÇÕES

Codec – é o acrônimo de Codificador/Decodificador, dispositivo de *hardware* ou *software* que codifica e decodifica sinais. No caso do presente trabalho, codificador/decodificador de vídeo. Por exemplo: MPEG, Xvid, DivX, RMVB, WMV dentre outros.

API – Abreviação em inglês de *Application Programming Interface* (ou Interface de Programação de Aplicativos) é um conjunto de rotinas e padrões estabelecidos por um *software* para utilização de suas funcionalidades por programas aplicativos. API's são como bibliotecas de funções específicas para certas aplicações específicas.

DirectX – Ou Microsoft DirectX© é uma coleção de API's que tratam de tarefas relacionadas a programação multimídia, em especial jogos para o sistema operacional Microsoft Windows©, ou seja, é quem padroniza a comunicação entre *software* e *hardware* e interpreta as instruções gráficas.

Bitmap – Imagens *raster* (ou *bitmap*, que significa mapa de bits em inglês) são imagens que contém a descrição de cada *pixel*, em oposição aos gráficos vetoriais.

Metadado – É o conjunto de caracteres alfanuméricos expresso, geralmente, por um esquema de uma base de dados orientada a objeto ou relacional (GUPTA; JAIN, 2007). Para alguns, metadado é simplesmente uma informação textual (dado) que descreve outro dado.

CBIR – *Content Based Image Retrieval* ou Recuperação de imagem com base em conteúdo.

## 2 ANÁLISE DE CONTEÚDO DE VÍDEO

Nos dias atuais, os avanços tecnológicos nas áreas de captura, armazenamento e transferência de dados tiveram como conseqüência a produção de um vasto acervo de conteúdos multimídia. Entretanto, interagir com conteúdos multimídia requer muito mais que capacidade de processamento e armazenamento em larga escala e banda nas redes de telecomunicações que levam esses conteúdos aos consumidores. Isto porque, os sistemas para organização, descrição e gerenciamento do conteúdo multimídia ainda são limitados e imprecisos, dificultando as operações de recuperação desses conteúdos pelos usuários.

Um vídeo digital é um caso específico de conteúdo multimídia que apresenta problemas similares aos apresentados. A solução para tais problemas passa, necessariamente, pela análise de como esses conteúdos estão estruturados (DIMITROVA *et al*., 2002). Analisar o conteúdo de um vídeo significa entender a semântica desse conteúdo, do ponto de vista computacional (WANG; LIU; HUANG, 2000). Para dar suporte à análise e indexação automática dos vídeos, faz-se necessário o emprego de ferramentas de análise das imagens, com o objetivo de segmentar os vídeos e extrair destes, características que descrevam matematicamente o seu conteúdo. Uma vez extraídas essas características, a interpretação automática e identificação do conteúdo semântico permite a construção de um índice, que será usado para realização de operações de recuperação baseadas em conteúdo. De acordo com Zhang *et al.* (1994), a indexação é um mecanismo que dá suporte à busca eficiente numa coleção de vídeos. Esse índice é construído com base em características intrínsecas dos dados do vídeo e/ou de seu conteúdo semântico.

O conteúdo visual do vídeo também pode ser descrito tendo como base outras características, tais como o tipo de evento, objetos e ações contidos nos mesmos. Seja qual for o tipo de informação utilizada para descrever seu conteúdo, é necessário definir-se um modelo de representação da informação contida no vídeo. Duas técnicas de modelagem do conteúdo de um vídeo são geralmente aplicadas: **estratificação** e **segmentação** (GUAN; KUNG; LARSEN, 2000). Como será mostrado mais adiante, o presente trabalho tem foco numa modelagem baseada em segmentação.

## 2.1MODELAGEM POR ESTRATIFICAÇÃO

A técnica de estratificação está mais centrada na segmentação conceitual da informação em blocos (*chunks*) do que na divisão física de quadros contínuos em tomadas, como na modelagem por segmentação. Na estratificação, cada bloco (conhecido como *stratum*) funciona como uma camada, cujo objetivo principal é de descrever a ocorrência de um simples evento (DAVENPORT; SMITH; PINCEVER, 1991). É possível citar como exemplo, a ocorrência de um evento específico tal como o trecho em que o âncora do tele-jornal apresenta uma notícia e assim por diante. Cada ocorrência é então modelada na linha do tempo como sendo um *stratum*. Na modelagem por estratificação, qualquer instante do vídeo pode ser descrito pela união dos fragmentos de *strata* presentes num dado momento, como uma linha vertical que intercepta um ou mais *stratas* num instante da linha do tempo (horizontal) (CHUA; CHEN; WANG, 2002). A Figura 1 ilustra um exemplo de modelagem de vídeo por estratificação com a ocorrência dos objetos/eventos no tempo.

A principal vantagem da estratificação sobre a segmentação é o fato de ser possível automatizar o processo de indexação. Entretanto, para tal, é preciso que uma ferramenta reconheça automaticamente diversos tipos de *strata*, o que não é uma tarefa trivial. Outra possibilidade é a rotulação manual dos *stratas.* Contudo, essa tarefa é extremamente tediosa e custosa. Davenport, Smith e Pincever (1991) empregam o uso do modelo de estratificação para representar conteúdo de vídeo e recuperá-lo em múltiplos contextos.

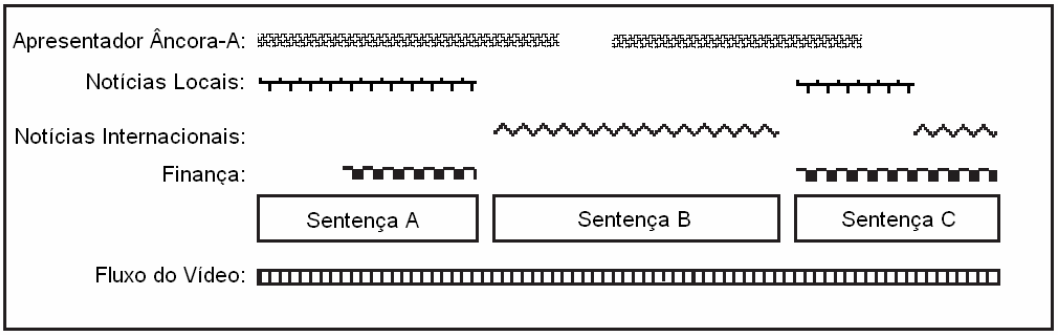


Figura 1 – Modelagem baseada em objetos de um vídeo de notícias.
Fonte: Guan, kung e Larsen, (2000).

## 2.2 MODELAGEM POR SEGMENTAÇÃO

Na modelagem por segmentação uma seqüência de vídeo é dividida em unidades atômicas, chamadas de **tomadas**. Essas unidades são fundamentais para o modelo de segmentação, uma vez que elas são utilizadas, posteriormente, para representação, análise e recuperação do vídeo. Importantes atributos também podem ser adicionados à tomada, tais como título, descrição, histograma de cor predominante nos quadros, objetos presentes e etc. Também podem ser adicionadas informações de câmera, tais como distância focal, tipo de ângulo, tipo de movimento, dentre outros, num processo, denominado pelas empresas de indexação de vídeos, de *logging* (GUAN; KUNG; LARSEN, 2000). Ao fim desse processo, as tomadas de vídeo "*loggadas*" são armazenadas para futuras consultas (CHUA; CHEN; WANG., 2002). As etapas de segmentação por tomadas e *logging* do vídeo fazem parte de um processo mais amplo chamado de *vídeo parsing*.

A etapa final para representação do vídeo com objetivo de dar suporte à recuperação e à navegação é chamada de indexação. O processo de indexação diz respeito à definição de uma estrutura de armazenamento dos segmentos extraídos, em conjunto com informações de conteúdo e contexto, numa base de dados. A recuperação de conteúdo e a navegação dependem fortemente do resultado dos processos de vídeo *parsing* e indexação do conteúdo de um vídeo (GUAN; KUNG; LARSEN, 2000). A Figura 2 ilustra um resumo das diversas etapas envolvendo os processos de representação e recuperação do conteúdo de um vídeo.

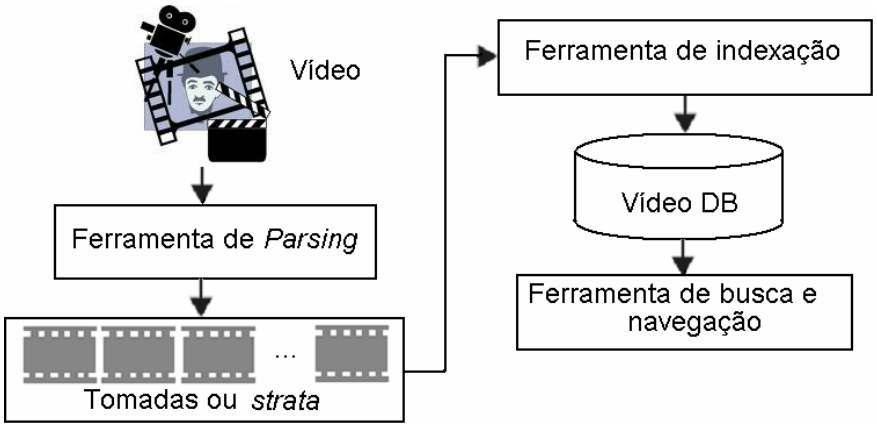


Figura 2 – Estrutura de indexação de vídeo por segmentação.
Fonte: Guan Ling *et al.*, (2000). (Adaptado)

## 2.3 ESTRUTURA DE VÍDEO

Um vídeo é uma seqüência ou um conjunto $V = \{q_1, q_2, q_3 \ldots, q_n\}$ composto de quadros $q_i$, que quando reproduzidos em determinada velocidade, apresentam a ilusão de movimento. Tipicamente, são usados 30 quadros por segundo[2] para se obter tal ilusão.

Um vídeo ainda contém normalmente um canal de áudio sincronizado à seqüência de quadros. Muito embora uma seqüência de vídeo possa não ter esse canal de áudio como no cinema mudo e em outros tipos de vídeos como na área de vigilância.

Existem várias terminologias para descrever vários atributos do vídeo. Nesse trabalho, procuram-se usar as terminologias mais comuns, no que se refere à imagem, ao vídeo e ao áudio, as quais são apresentados no Quadro 1, adaptada de Guan, Kung e Larsen, (2000). As outras notações utilizadas no presente trabalho são aquelas apresentadas no item 1.7.

Quadro 1 - Terminologia de vídeo

| **Termo** | **Comentário** |
|---|---|
| Vídeo (Imagem/Áudio) | O termo vídeo será usado para representar um fluxo de imagens e áudio. |
| Cena (Imagem) | Uma cena é uma seqüência de quadros delimitados no tempo que carregam uma unidade semântica. |
| Quadro ou *Frame* | Refere-se a um único fotograma do vídeo. |
| Segmento | Um subconjunto homogêneo de quadros do vídeo, delimitados no tempo por semântica ou não. |
| Tomada | Conjunto de quadros consecutivos entre um corte de câmera e outro. |
| Áudio | Refere-se ao(s) canal(is) de áudio associado(s) ao vídeo. |

Fonte: (GUAN; KUNG; LARSEN, 2000).

Em síntese, com relação à estrutura do vídeo, considera-se que a tomada consiste de um ou mais quadros cujas características (visuais) são semelhantes. Estes quadros são gerados ou filmados de forma contínua, representando uma ação em relação ao tempo e espaço. As cenas são uma combinação de uma ou mais tomadas dentro de um mesmo contexto ou semântica. Por sua vez, um conjunto de cenas formam um vídeo, como exibido na Figura 3 (SANTOS, 2004).

---

[2] É comum se encontrar o termo em inglês, FPS (*Frames Per Second*), para designar a taxa quadros por segundo.

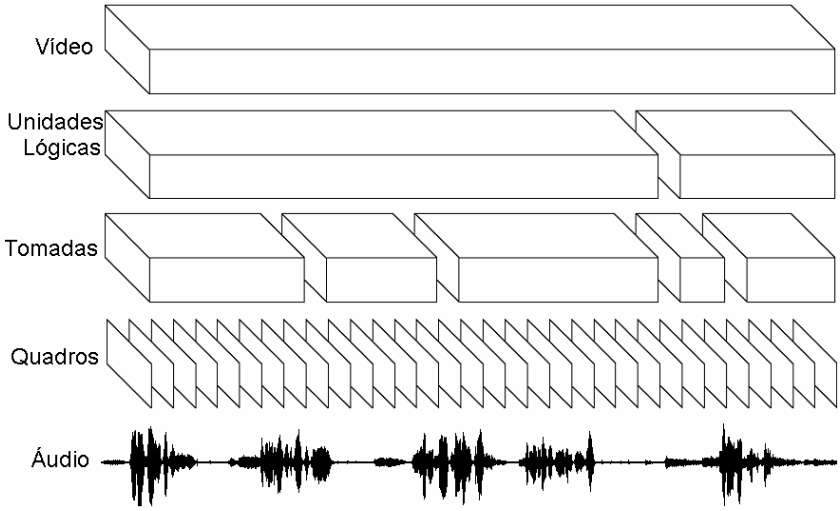


Figura 3 – Estrutura ou representação hierárquica do vídeo
Fonte: Sundrram *et al*., (2000) (Adaptado).

A trilha de áudio agrega uma grande quantidade de informação ao fluxo do vídeo e é de extrema importância na criação de filmes. No início, vídeos eram exibidos sem áudio por limitações de ordem tecnológica, posteriormente, foi adicionada música ao vivo junto à exibição do vídeo. Atualmente, as ferramentas evoluíram e a forma de criar vídeos também, de modo que, existem profissões especificas no cinema para cuidar da composição, da seleção e da edição do áudio (DAVENPORT; SMITH; PINCEVER, 1991). Devido à riqueza encontrada na trilha de áudio é possível usa-la na segmentação de cenas (SUNDARAM; CHANG, 2000), na segmentação de trechos onde há música, fala, ou música e fala simultâneas (PINQUIER; SÉNAC; ANDRÉ-OBRECHT, 2002) e na classificação do áudio (FOOTE, 1997). A análise do canal de áudio também pode ser utilizada para a classificação e recuperação de conteúdo com as técnicas de extração de características de áudio apropriadas. Entretanto, dado que o foco do presente trabalho está restrito ao processamento apenas do conteúdo visual de um vídeo, o restante do texto será dedicado às questões relacionadas à representação de tal estrutura de um vídeo.

### 2.3.1 Representação da Estrutura Visual de um Vídeo

Conforme a Figura 3, a estrutura visual do conteúdo de um vídeo pode ser representada, de forma hierárquica, através de quadros, tomadas e unidade lógicas ou cenas.

#### 2.3.1.1 Quadros

Um quadro (ou *frame*) $q_t$ pode ser descrito como sendo uma função bidimensional, na qual cada posição espacial $(x, y)$, $q_t(x, y)$ representa o valor de intensidade de luz do *pixel* $(x, y)$ num instante de tempo $t$. Visto individualmente, cada quadro $q$ é uma imagem digital $q(x, y)$ discretizada tanto em coordenadas espaciais, quanto em intensidade de brilho. Os elementos espaciais da imagem $(x, y)$ são chamados de *elementos de imagem, elementos da figura, "pixels"* ou *"pels"* (abreviação de *picture element*). Assim, um *pixel* é a menor unidade da imagem (GONZALEZ; WOODS, 2000). Visto individualmente, pode-se dizer que cada quadro do vídeo tem a mesma representação de uma imagem estática.

#### 2.3.1.2 Tomadas

De acordo com Davenport, Smith, e Pincever (1991), uma tomada (ou *shot*) é definida como uma seqüência de quadros gerados ou gravados continuamente e que, representam uma ação contínua no tempo e no espaço. Sendo uma tomada uma seqüência $S_i = \{q_{t+1}, q_{t+2}, q_{t+3}, \ldots, q_{t+n}\}$, considera-se um quadro $q_i$ como sendo a menor unidade de uma tomada. Assim definidas, as tomadas são os blocos básicos da constituição dos vídeos e portanto consideradas entidades físicas delimitadas por fronteiras de tomadas ou *shot boundaries* (RUI; HUANG; MEHROTRA, 1999).

Quanto à forma de detecção automática de tomadas, as técnicas geralmente usadas podem ser classificadas em cinco categorias: baseada em *pixel*; baseada em estatística; baseada em transformada; baseada em características visuais e baseada em histograma. Alguns pesquisadores alegam que a técnica baseada em histograma apresenta os melhores resultados, a exemplo de Rui, Huang e Mehrotra (1998).

#### 2.3.1.3 Unidades lógicas de vídeo

As tomadas podem ser agrupadas de modo a se obter segmentos homogêneos de um vídeo. Diferentemente da segmentação de "baixo-nível" ou por tomadas, na qual apenas os cortes físicos de câmera são levados em conta, na segmentação de "alto nível" ou homogênea, os agrupamentos são definidos a partir de informações lógicas ou semânticas associadas ao conteúdo. Segmentar o vídeo em tomadas é uma etapa necessária e anterior à segmentação de alto-nível. Nessa última, as estruturas sem contexto (tomadas) são agrupadas para formar unidades

de vídeo com estrutura semântica. Esses segmentos são chamados de “unidades lógicas de vídeo” *Logical Story Unit (LSU*).

Uma *LSU* pode ser entendida como uma representação aproximada de um trecho de vídeo, que é caracterizado por um evento simples (tal como um diálogo, uma cena de ação, etc) ou por uma série de eventos semanticamente relacionados que acontecem paralelamente durante esse trecho. Uma vez considerado o evento semântico como um todo (e não uma tomada) como sendo a unidade mais natural para filmes e programas, a segmentação em unidades lógicas passa a ser muito importante para sistemas que dão suporte à navegação e à recuperação baseadas em conteúdo (HANJALIC; LAGENDIJK; BIEMOND, 1999).

Os autores Hanjalic, Lagendijk e Biemond (1999) definem uma *LSU* como sendo um segmento consistente em termos temporais e visuais. Com relação à consistência, pode-se esperar que um evento esteja relacionado com elementos visuais e objetos característicos de uma cena que representam o mundo real, tais como: cenário, pessoas, objetos, fundos, faces, roupas e outros padrões característicos. Assim, os autores finalizam a definição de uma *LSU*, considerando-a como uma série de tomadas contíguas que se conectam por similaridade visual ou semântica.

De acordo com o tipo do vídeo ou do ponto de vista adotado, uma unidade lógica pode ser classificada em diferentes subcategorias semânticas, tais como cenas, diálogos, tópicos e outros.

#### 2.3.1.4 Cenas

Uma cena é uma unidade lógica tipicamente utilizada para representar a estrutura de vídeos de ficção, tais como filmes, novelas e seriados de TV. Uma cena[3] de vídeo consiste numa seqüência de tomadas semanticamente relacionadas (LIN; ZHANG, 2000). Uma vez que máquinas não compreendem o contexto de vídeos, a segmentação automática eficiente de cenas é uma das tarefas mais complexas para a estruturação de um vídeo.

#### 2.3.1.5 Diálogos

Um diálogo é uma unidade lógica caracterizada pela conversação entre personagens de vídeo de ficção, entrevistas e outros segmentos temporais que

[3] Em inglês, também se encontra o termo *Story Unit* referindo-se a cena.

envolvem duas ou mais pessoas. Os autores Lehane, O'connor e Murphy (2004) apresentam uma solução para detecção de diálogos em filmes utilizando características visuais. Tipicamente, verifica-se a re-ocorrência de quadros-chave com similaridade visual. Por exemplo, em um dialogo entre duas pessoas, será comum corte de tomadas com quadros-chaves na seqüência A-B-A-B-A-B.

#### 2.3.1.6 Tópico

O tópico é uma unidade lógica geralmente associada ao noticiário, documentário e vídeos educativos. Nessa categoria, um tema ou tópico específico é abordado e discutido no vídeo. Na literatura, também são encontradas outras categorizações de unidades lógicas, tais como episódios (HANJALIC; LAGENDIJK; BIEMOND, 1999), parágrafos de vídeo e macros segmentos (TRUONG, 2004).

## 2.4 SEGMENTAÇÃO AUTOMÁTICA DE TOMADAS

A delimitação de tomadas (ou *shots*) de vídeo é um dos conceitos base para a indexação e estruturação do conteúdo de vídeos, agregando quadros contíguos em seqüência com o mesmo contexto (BOVIK; GIBSON, 2000).

Os cortes de tomada podem ser definidos pelos produtores de vídeo com ou sem a utilização de efeitos de transição. No corte simples ou abrupto, não há efeitos especiais entre uma tomada e outra, o intervalo de tempo entre as tomadas é o mínimo. Quando algum efeito é adicionado na fronteira da tomada, o corte ocorre através de uma **transição gradual** ou, simplesmente, **transição**. Normalmente as transições consistem na adição de uma série de quadros artificiais gerados por uma ferramenta de edição (GUIMARÃES, 2003). As transições graduais são principalmente de três tipos: *fade*, *dissolve* ou dissolução e *wipe*.

O efeito de *fade* consiste na progressiva transição da tomada para uma única tonalidade de cor, geralmente a tonalidade preta e vice-versa. O efeito de *fade* pode ser subdividido em *fade-out* e *fade-in*. No *fade-out* há um progressivo desaparecimento do conteúdo visual para o quadro em mono-cor. Já o efeito de *fade-in* é caracterizado pelo aparecimento progressivo do conteúdo visual a partir de um quadro monocromático (LIENHART; KUHMÜNCH; EFFELSBERG, 1997).

A dissolução é caracterizada pela transição de duração não nula e progressiva da tomada $S_t$ para a tomada consecutiva $S_{t+1}$ (GUIMARÃES, 2003). Na

dissolução há um *fade-out* em $S_t$ ocorrendo simultaneamente com um *fade-in* em $S_{t+1}$ (SANTOS, 2004).

O *wipe* é um efeito óptico em que o quadro final da tomada $S_t$ é sucessivamente trocado pelo quadro inicial da tomada $S_{t+1}$ com algum efeito de deslizamento, compressão, abertura ou outro efeito espacial de troca entre os quadros de fronteira. A imagem da tomada seguinte pode aparecer deslizando de uma borda para outra, surgir no meio ou em qualquer outro ponto da imagem e gradualmente tomar a tela, isso seguido por algum padrão geométrico (TRUONG, 2004).

Com relação às transições, elas ainda podem ser categorizadas de acordo com a sua classe. Na **transição cromática**, apenas o espaço de cor dos quadros é manipulado e a duração da transição não é nula. Encaixam-se nas transições cromáticas os efeitos de *fade* e dissolução. Já a **transição espacial** atua no aspecto espacial das tomadas, tal como no efeito *wipe*. A duração da transição espacial é não nula, entretanto, a duração da transformação de *pixel* a é. Finalmente, é possível uma combinação das transições cromáticas e espaciais, produzindo uma **transição espacial-cromática**, quando ambos os efeitos ocorrem simultaneamente (GUIMARÃES, 2003).

## 2.5 SEGMENTAÇÃO DE UNIDADES LÓGICAS

A segmentação automática de unidades lógicas têm sido explorada por diversos autores (HANJALIC; LAGENDIJK; BIEMOND, 1999; LIN; ZHANG, 2000; LEHANE; O'CONNOR; MURPHY, 2004; SUNDARAM; CHANG, 2000; BORECZKY; WILCOX, 1998; RUI; HUANG; MEHROTRA, 1998) para diferentes categorias de vídeos. Devido às particularidades envolvidas nessas categorias, diferentes técnicas de segmentação automática são empregadas. Por exemplo, a segmentação de unidades lógicas em vídeos de noticiários são feitas levando-se em consideração a ocorrência de tomadas nas quais o apresentador (âncora) do tele-jornal aparece. Assim, algoritmos desenvolvidos para a categoria de vídeos de noticiários podem não funcionar adequadamente em filmes, novelas ou outra categoria particular de vídeo.

Alguns autores utilizam características da trilha de áudio em conjunto com características visuais com o objetivo de obter melhores resultados na segmentação de cenas (SUNDARAM; CHANG, 2000; BORECZKY; WILCOX, 1998). Sundaram e

Chang (2000), por exemplo, partem da idéia de que uma cena é um bloco de dados áudio-visual consistente. Esses autores utilizam um modelo de memória para realizar uma segmentação baseada em características de vídeo e outra baseada em características de áudio. O algoritmo que cuida da segmentação baseada na trilha de áudio, extrai dez características, tais como energia, taxa de cruzamento de zeros, dentre outras. Essas características são extraídas em janelas de 100ms de duração e armazenados. Já o algoritmo de segmentação visual, determina a coerência entre quadros-chave extraídos das tomadas e armazenados na memória. Numa última etapa, um algoritmo computa a coerência entre as duas segmentações apresentando uma segmentação final.

Os autores Boreczky e Wilcox (1998) extraem características de distância entre quadros com base em histograma, distâncias por características de áudio e estimativa de movimento entre os quadros. Todos esse elementos são combinados num modelo de Cadeia Ocultas de Markov ou *Hidden Markov Model* para treinar e segmentar unidades lógicas de vídeo. Essa abordagem elimina dois problemas comuns. O primeiro é a determinação de limiares para detecção de cortes quadro-a-quadro e distâncias para detecção de cortes graduais, já que com o uso de Cadeias de Markov esses parâmetros são “aprendidos” automaticamente. O segundo problema é o de como usar múltiplas características, como diferenças de histogramas, vetores de movimento e características de áudio na segmentação do vídeo. As Cadeias de Markov permitem que qualquer tipo de característica seja adicionada ao seu vetor.

Outros trabalhos realizam a segmentação do conteúdo através do processo de inferência por similaridade (*similarity linkage inference*). Os autores Hanjalic, Lagendijk e Biemond (1999) aplicam a idéia de que a similaridade visual entre as tomadas do vídeo podem ser medidas. Desse modo, constrói-se uma estrutura de representação do conteúdo visual contendo as características visuais das tomadas e conectam-se as tomadas visualmente similares como visto na Figura 4.

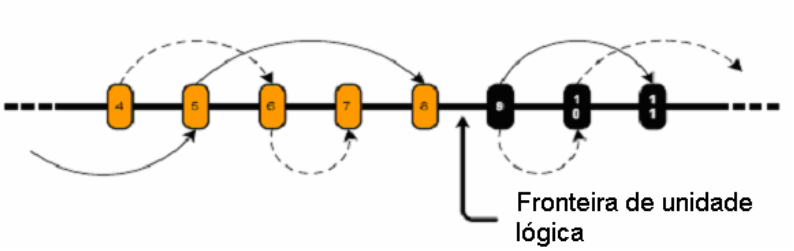


Figura 4 - Similarity linkage inference

Fonte: Truong (2004).

A partir da similaridade, são criados apontadores ou *links* entre tomadas com características em comum. A repetição do padrão de uma determinada tomada após uma outra diferente é um indicativo de continuidade semântica, ou seja, a unidade lógica, ainda não foi cortada. Nesse caso, os elementos em comum estão se repetindo de forma intercalada até o momento em que há uma quebra no padrão e, por sua vez, uma quebra nos apontadores ligando tomadas similares. Essa quebra pode ser entendida como um indício de que há uma fronteira semântica ou uma nova unidade lógica.

## 2.6 SUMARIZAÇÃO DE VÍDEO

Assim como a indústria do cinema produz *trailers* de filmes para atrair espectadores, as emissoras de TV apresentam resumos de sua programação durantes os intervalos comerciais dos seus programas. Com o grande volume de vídeos sendo gerados, torna-se financeiramente proibitivo editar e gerar resumos para todo e qualquer vídeo produzido pelas emissoras e é exatamente para preencher essa lacuna que surge a sumarização automática de vídeos.

De acordo com Pfeiffer *et al*., (1996), o **sumário de vídeo** é definido como uma seqüência de imagens estáticas ou em movimento representando o conteúdo de um vídeo. Essa representação resumida do conteúdo permite que o espectador assimile rapidamente alguma informação concisa sobre o conteúdo completo do vídeo, enquanto a mensagem original é preservada[4].

Considerando uma taxa de 30 quadros por segundo, em apenas uma hora de vídeo, existem cerca de 108 mil quadros. Essa grande quantidade de quadros associada a um vídeo dificulta tarefas como a anotação de seu conteúdo, a sua navegação e ainda, a extração automática de características. Além do mais, quadros

[4] Na literatura científica não há um consenso com relação à definição de “sumário de vídeo”.

consecutivos tendem a ser visualmente muito similares, apresentando uma forte redundância.

Representar uma seqüência de vídeo de modo resumido ou sumarizado é útil para uma variedade de aplicações. A sumarização de vídeo (*video abstraction*) provê uma forma rápida de acesso ao conteúdo de vídeo num sistema de recuperação, além de permitir uma rápida visão geral do conteúdo do vídeo (HANJALIC; ZHANG, 1999). Existem basicamente duas formas de sumarização de vídeo: por *video skim* e por quadros-chave ou representativos (*key-frames*) (TRUONG, 2004).

O *video skim*, também conhecido como *moving-image abstract* ou *moving story board*, é uma sumarização formada por segmentos importantes do vídeo (com seu respectivo canal de áudio), de modo a resumir o conteúdo como um todo. Os *trailers* de filmes são exemplos típicos desse tipo de sumarização.

O *key-frame,* também conhecido como *representative frame*, *r-frames*, *still-image abstract*, *static storyboard*, ou ainda, em português quadro-chave, é um quadro extraído do vídeo a partir de um determinado critério a fim de caracterizar parte do seu conteúdo.

Uma vantagem da sumarização de vídeo no formato *skim* é a possibilidade de adicionar áudio e movimento enriquecendo a abstração do resumo. Além do que, é mais interessante ver um *trailer* que uma seqüência de quadros estáticos.

A extração de quadros-chave é mais adequada para navegação em vídeo e recuperação de conteúdo visual, onde os usuários podem se guiar e selecionar diretamente o segmento de interesse do vídeo. Outro ponto importante no uso de quadros-chave é a aplicação dos mesmos como base para extração de características e indexação de vídeo para futura recuperação (ZHANG; WANG; ALTUNBASAK, 1997). Esse, por sinal, é o grande foco da presente pesquisa. Outra vantagem dos quadros-chave é que os mesmos não requererem sincronização, uma vez extraídos, é possível reorganizá-los seguindo qualquer critério para navegação. Essa é uma vantagem importante para a recuperação de vídeo baseada em conteúdo. No caso dessa dissertação, os quadros-chave resultantes de uma consulta são organizados por ordem de similaridade visual.

Embora os processos de geração de quadros-chave e *video skim* sejam diferentes, é possível gerar quadros-chave a partir de um *video skim*. Ou, no sentido inverso, unir um conjunto de quadros-chave capturados de modo a gerar um resumo

de vídeo, no qual também é possível adicionar uma trilha de áudio (TRUONG, 2004). Nesse último caso, um *video skim* gerado a partir de uma série de quadros-chave pode ser semelhante à visualização do vídeo em alta velocidade ou modo *fast-preview* (PFEIFFER *et al*., 1996) como é conhecido nos sistemas de vídeo baseados em fita. Esse modo de sumário, *video skim* em *fast-preview*, dificulta a navegação no vídeo, já que a alta velocidade com que as cenas são exibidas não permite que o ponto exato desejado seja corretamente selecionado. Um outro problema é que, como uma seqüência em *fast-preview* é exibida em alta velocidade, a redundância temporal é reduzida e, por conseguinte, a taxa de compressão do sumário. Uma última dificuldade fica por conta da trilha de áudio, pois perde o sentido com a alta velocidade do vídeo.

Quanto aos métodos de sumarização de vídeo, estes podem ser feitos de três modos:

a) **Manual:** o material do sumário é completamente escolhido por humanos;
b) **Semi-automático:** a seleção inicial do material é feita por um computador, posteriormente alguém avalia e decide sobre o sumário final;
c) **Automático:** o sumário é completamente gerado por computador.

O presente trabalho apresenta uma forma de sumarização automática do vídeo no formado *static storyboard*, ou quadros representativos. Este sumário tem um papel fundamental na indexação e recuperação de quadros dos vídeos, de modo que, o sucesso das buscas está em parte relacionado com a qualidade do resumo produzido.

## 2.7 EXTRAÇÃO AUTOMÁTICA DE QUADROS-CHAVE E DETECÇÃO DE CORTES DE TOMADA

Nessa seção serão discutidos os métodos empregados no presente trabalho para a detecção de corte de tomada e extração de quadros-chave.

Como se sabe, o conteúdo de um vídeo possui grande redundância temporal. Essa redundância já foi explorada nos codificadores de vídeo com o objetivo de comprimir o conteúdo, a exemplo do MPEG-2. Pelo mesmo motivo de redundância temporal, no presente trabalho, apenas um percentual dos quadros do vídeo são processados e têm suas características extraídas para posterior recuperação. Na

presente pesquisa, a extração de quadros-chave é usada tanto na representação do sumário do vídeo para visualização quanto para extração de características visuais empregadas na indexação.

### 2.7.1 Métodos para detecção automática de corte de tomada e extração de quadros-chave

Em sua essência, pode-se afirmar que os métodos para detecção automática de corte de tomada e para extração de quadros-chave utilizados nesse trabalho são basicamente os mesmos. A diferença principal está nos limiares aplicados e nos canais de cores empregados em cada método. Outra diferença relevante é que, considerando $q_i$ um quadro no instante $i$, para a detecção de corte de tomada, faz-se uma comparação entre quadros adjacentes $q_i$ e $q_{i+1}$, enquanto extração de quadros-chave verifica-se a diferença entre o quadro-chave $i$ e os próximos $n$ quadros candidatos a novo quadro-chave $q_i$ e $q_{i+n}$.

Estas técnicas têm sido utilizadas há algum tempo na detecção de corte de tomada, uma vez que se espera que fronteiras de tomadas apresentem uma distância entre os quadros $q_i$ e $q_{i+1}$ acima de um certo limiar (SANTOS, 2004), a mesma idéia pode ser empregada para a eleição de quadros-chave.

#### 2.7.1.1 Comparação de quadros *pixel-a-pixel* ou distância entre quadros

A abordagem mais simples e direta para comparação de quadros é o cálculo da distância entre eles a partir dos valores dos seus *pixels* ocupando a mesma posição espacial. Para isso, pode-se utilizar a norma $L^1$ ou $L^2$ (respectivamente eq. 1 e eq. 2)

$$\|f_k, f_l\| = \sum_{i,j} |f_k[i,j] - f_l[i,j]| \qquad \textbf{eq. 1}$$

$$\|f_k, f_l\|_2 = \sqrt{\sum_{i,j} (f_k[i,j] - f_l[i,j])^2} \qquad \textbf{eq. 2}$$

Sendo: $f$ a representação de um quadro, $k$ e $l$ a posição temporal do quadro e $i$ e $j$ a posição espacial dos seus *pixels*.

Embora as eq. 1 e eq. 2 não deixem explícito o canal de cor utilizado na comparação, essas equações obviamente valem para qualquer canal de cor. É importante frisar que os quadros comparados precisam ter a mesma resolução

espacial, o que não é um problema nos vídeos, já que todos os quadros têm a mesma resolução.

Essas abordagens de comparação de distância têm sido descritas para detecção de cortes de tomadas uma vez que espera-se que fronteiras entre tomadas apresentem uma distância entre os quadros $q_i$ e $q_{i+1}$ acima de um certo limiar. Um dos principais problemas do uso da diferença *pixel*-a-*pixel* para detecção de cortes de tomadas é que a métrica é intolerante à movimentação brusca de objetos e de câmera (SANTOS, 2004).

Para a seleção de quadros-chave, o mesmo método pode ser empregado. O primeiro quadro do vídeo, $q_0$, é também o primeiro quadro selecionado como chave. Utilizando o último quadro-chave selecionado como referência, calcula-se a distância entre os quadros seguintes do vídeo até que um limiar seja ultrapassado. Esse novo quadro é então eleito como quadro-chave. O pseudocódigo do método é mostrado no Algoritmo 1.

```
1  i = 0;
2  qChave = q[i];
3  for (i = 1; i<totalQuadros; i++)
4     if (dist(qChave, q[i]) > limiar)
5         qChave = q[i];
6     end
7  end
```

Algoritmo 1 – Seleção de quadros-chave por distância entre quadros

A seleção ideal de quadros-chave depende dos objetivos da aplicação. A seleção para sumarização pode variar de acordo com o nível de detalhamento do sumário desejado, tendo a perda de quadros uma relação direta com a tolerância da quantidade de quadros que representem tomadas utilizadas no sumário. Especificamente neste trabalho, e como será mostrado em momento oportuno, o objetivo da aplicação foi selecionar quadros-chave para sumarizar o vídeo e permitir a busca posterior com base nas características visuais destes quadros-chave.

#### 2.7.1.2 Comparação de quadros por histograma de ocorrência

Para a comparação de quadros por histograma, primeiramente, extrai-se o histograma de ocorrência dos *pixels* dos quadros a serem comparados. Uma das técnicas mais utilizadas para detecção de corte de tomada é o uso de histogramas de intensidade luminosa. O histograma de uma imagem digital com níveis de cinza no intervalo [0, $L$-1] é uma função discreta $p(r_k) = n_k / n$, onde $r_k$ é o *k*-ésimo nível de cinza, $n_k$ é o número de *pixels* na imagem com ocorrência do valor de intensidade $k$,

$n$ é o número total de *pixels* na imagem e $k$ = 0,1,2,..., $L$–1 (GONZALEZ; WOODS, 2000).

$$z_n(f_i, f_j) = \sum_{l=o}^{L-1} \left| p_i[l] - p_j[l] \right| \qquad \text{eq. 3}$$

Na equação, $f_i$ e $f_j$ denotam, respectivamente, o quadro $i$ e $j$; $p_i$ e $p_j$ denotam respectivamente os histogramas dos quadros $i$ e $j$; e $z_n(f_i, f_j)$ denota a função de distância entre os dois histogramas.

Embora a comparação de quadros por histograma tenha sido apresentada em termos de imagens monocromáticas, o mesmo princípio pode ser aplicado em imagens no domínio RGB ou qualquer outro domínio espacial de cor.

Essa técnica também pode ser utilizada na seleção de quadros-chave. Alterando-se apenas a função de distância *pixel*-a-*pixel* para a função de distância de histogramas no Algoritmo 1. Não são necessárias outras alterações.

## 2.8 DESCRIÇÃO DE CONTEÚDO

A descrição de conteúdo multimídia pode ser feita de diversas maneiras. Um padrão utilizado para isso é o MPEG-7, que possui como vantagem a flexibilidade quanto ao tipo de conteúdo (vídeo, imagem, áudio, etc.) que pode ser descrito. Desse modo, tanto humanos quanto sistemas automáticos podem usar os padrões para descrição de conteúdo audiovisual e armazená-los no formato MPEG-7 (MARTÍNEZ, 2007). Os metadados de natureza automática são tipicamente características de baixo nível (EIDENBERGER, 2003) enquanto que as informações inseridas manualmente podem carregar conteúdo de valor semântico, por isso são chamadas de características de alto nível (CANTARELLI; SOTT, 2006; KIM *et al.*, 2003; TSINARAKI; CHRISTODOULAKIS, 2005). Como exemplo de características de alto nível podemos citar: o nome dos atores que participam de uma cena, os objetos e o ambiente do vídeo. Já as características de baixo nível que podem ser extraídas são automáticas e baseadas em métodos matemáticos e estatísticos. O capítulo seguinte é justamente direcionado a esse tipo de característica, que é de grande importância no suporte à recuperação baseada no conteúdo visual analisado nesse trabalho.

# 3 EXTRAÇÃO DE CARACTERÍSTICAS DE BAIXO NÍVEL EM VÍDEO

O presente capítulo apresenta alguns fundamentos ligados às características de baixo nível (ou características primitivas) que podem ser extraídas a partir do processamento do conteúdo visual de um vídeo. Estas características são fundamentais para o suporte às diferentes formas de recuperação de um vídeo com base no conteúdo visual.

## 3.1FUNDAMENTOS DA BUSCA DE QUADROS E IMAGENS

Dois tipos de informação são associados aos objetos visuais (de uma imagem ou de um vídeo): a informação sobre os objetos, chamada de metadado, e a informação contida nos objetos, chamada de característica visual. Um metadado pode ser definido como uma entidade alfanumérica e geralmente expressa como um esquema relacional de uma base de dados. As características visuais são obtidas através de processos computacionais ligados ao processamento de imagem, computação visual e rotinas geométricas executadas sobre o objeto visual (GUPTA; JAIN, 2007). Essas características são normalmente subdivididas em três classes: características de cor; forma e textura (FENG; SIU; ZHANG, 2003).

A idéia básica por trás das técnicas de extração de características visuais é evitar a dependência da intervenção apenas humana no processo de descrição do conteúdo. Com isso, o conteúdo passa a ser representado por um conjunto de características que permitem que os usuários encontrem imagens “visualmente similares” a partir de uma determinada imagem de busca.

A similaridade visual está relacionada com a percepção que o usuário tem da imagem (por exemplo, se uma imagem é escura ou clara, se a mesma tem uma predominância de verde, vermelho ou azul, se está em níveis de cinza ou colorida, dentre outras características). É importante frisar que o termo “similar” pode ter significados diferentes sob a perspectiva de quem vê. Em outras palavras, pode-se dizer que ainda não é possível se definir um método único e genérico para caracterizar o conteúdo visual de um vídeo e que deve ser buscado um conjunto próprio de características específicas para cada perspectiva (DESELAERS, 2003).

Com isso, um dos objetivos principais deste trabalho é apresentar uma solução para recuperação de quadros de vídeo com semelhança visual ao de uma

"imagem exemplo", que pode ser uma imagem proveniente de captura do mundo real, ou ainda a partir de uma "imagem rascunho" (*sketch*) feita manualmente.

Conforme a Figura 5, a recuperação de imagem baseada em conteúdo é feita basicamente de três formas: **"busca por texto"**; **"busca por exemplo"** e **"busca por rascunho"** (DESELAERS, 2003).

Antes de descrever cada forma de busca, fica conceituado como sendo **imagem de busca *Q*** *(Query)* a imagem fornecida ao sistema de busca como "gabarito" ou "amostra" do que se espera recuperar por semelhança visual. E **Imagem alvo *T*** *(Target)*, ou simplesmente **alvo**, é o quadro ou são os quadros do vídeo que se pretendem recuperar com base na imagem de busca *Q*.

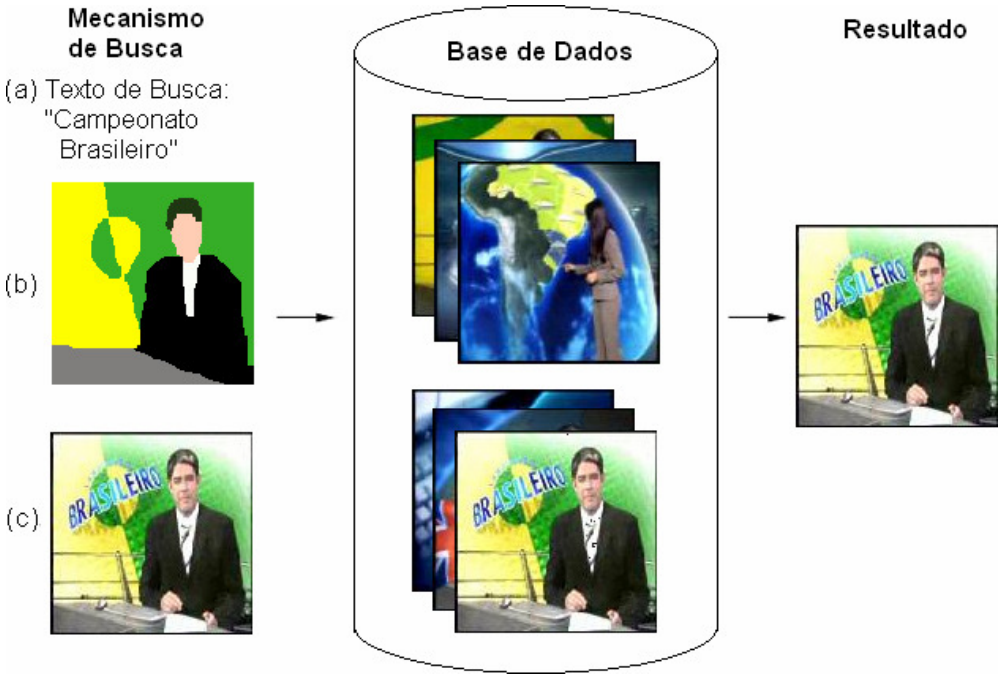


Figura 5 Busca de imagem - Busca por Texto (a), Busca por Rascunho (b), Busca por Exemplo (c).

Fonte: elaborada pelo autor (2008).

### 3.1.1 Busca por texto

Na busca por texto (ou baseada em texto), o usuário fornece uma descrição textual do conteúdo que procura, a qual é utilizada por um SGBD para recuperar o conteúdo. Nessa abordagem, é necessário que o conteúdo da imagem ou do vídeo seja previamente anotado seguindo algum padrão de metadados, por exemplo, o MPEG-7 (SANTOS; REHEM NETO, 2004; GERTZ *et al.*, 2002). Essa abordagem é

a mais eficiente do ponto de vista semântico. Entretanto, ela é bastante dependente das anotações feitas a *priori*, as anotações ficam restritas a um vocabulário (que ainda pode variar) e não permite a busca nas propriedades visuais da imagem. Além disso, a anotação manual é bastante custosa, pois, cada hora de vídeo anotado adequadamente consome cerca de dez horas de trabalho (IBM..., 2007). Para Dimitrova *et al.* (2002), além do custo natural da geração de uma anotação manual, quando é possível de ser realizada, ela é subjetiva, ineficaz e incompleta. Atualmente, podem ser encontradas ferramentas de anotação automática, contudo, apesar de vários esforços, estas ferramentas ainda não são suficientemente confiáveis (JEON; LAVRENKO; MANMATHA, 2003).

### 3.1.2 Busca por exemplo

Na “busca por exemplo”, o usuário oferece uma imagem de busca similar à que ele procura, essa imagem serve como uma espécie de “gabarito”, exemplo ou modelo do qual são extraídas características básicas utilizadas no processo de busca. Daí, o nome “busca por exemplo”. A imagem de busca pode ser uma fotografia, uma imagem em baixa resolução, um *thumbnail* semelhante ao quadro alvo desejado ou uma imagem sintética gerada por computação gráfica.

### 3.1.3 Busca por esboço ou rascunho

Na “busca por esboço” ou “rascunho”, o usuário faz um *sketch* da imagem que está procurando da melhor forma como ele lembra da imagem alvo, tão similar quanto possível ao quadro alvo em termos de cor, forma e posição dos objetos. Para realização da “busca por esboço”, no presente trabalho, são utilizadas características extraídas através da transformada *wavelet*, como será mostrado mais adiante.

Neste trabalho, serão consideradas as formas de busca de imagem “por exemplo” e “por rascunho”, conforme métodos de extração de características mostrados no capítulo 5 e os resultados apresentados no capítulo 6.

## 3.2 FUNDAMENTOS DE CARACTERÍSTICAS DE IMAGEM

A idéia básica por trás da **recuperação de imagem baseada em conteúdo** é permitir ao usuário encontrar imagens a partir de características visuais semelhantes. A seleção de quais características usar, entretanto não é uma tarefa

trivial. Isto ocorre porque o conceito de similaridade pode variar de uma pessoa para outra e porque diferentes características referem-se a diferentes propriedades da imagem.

As características visuais podem ser classificadas como locais ou globais. Elas são ditas locais quando as mesmas são extraídas a partir de um segmento da imagem original. Esse segmento ou subimagem pode ser uma janela quadrada de 15x15, 17x17 *pixels* ou outra resolução. As características são ditas globais quando extraídas considerando-se todos os *pixels* da imagem original (DESELAERS, 2003). Um exemplo de uso de características locais pode ser encontrado no trabalho (PIMENTEL FILHO; MONTALVÃO; REHEM NETO, 2006) que emprega essa técnica na classificação de textura de imagens.

Uma característica também pode ser classificada como invariante quando a mesma é mantida inalterada quando uma certa transformação é aplicada à imagem. Exemplos dessas transformações são: a rotação, a translação e o escalamento (DESELAERS; KEYSERS; NEY, 2004).

A abordagem mais direta para comparação da similaridade (ou diferença) entre imagens baseia-se nos valores dos *pixels* que as compõem. Um exemplo de abordagem é o escalamento das imagens para um mesmo tamanho (em termos de *pixels*), com a comparação da distância Euclidiana entre as mesmas, *pixel*-a-*pixel*. No presente trabalho, essa mesma técnica foi usada para dar suporte à detecção automática de corte de tomadas, enquanto que diversas outras características estatísticas (descritas posteriormente) foram usadas para a recuperação de quadros visualmente similares.

As características visuais são derivadas de processos computacionais, tipicamente, processamento de imagens, computação visual e rotinas geométricas executadas sobre o objeto visual (GUPTA; JAIN, 2007). As abordagens baseadas nas características visuais já demonstraram ter um bom desempenho quando utilizadas em reconhecimento óptico de caracteres e imagens médicas. Dentre as características que representam estatisticamente um conteúdo de um vídeo (ou imagem), podem ser destacadas as diversas informações ligadas à cor de cada *pixel*.

### 3.2.1 Cor

A cor é uma das características mais usadas em recuperação de imagens baseada em seu conteúdo. Os espaços de cor, que são tridimensionais, tornam a discriminação da imagem potencialmente superior as imagens P&B. Apesar de existirem diversos espaços de cor, tais como *Munsell*, CIE L*u*v, CIE L*a*b, YIQ e HSV (FENG; SIU; ZHANG, 2003), nesse trabalho estão sendo adotados os espaços RGB e YIQ apenas. Como não há um consenso a respeito de qual espaço é melhor para busca de imagens, o YIQ foi escolhido para as buscas das imagens em conformidade com o trabalho de Jacobs, Finkelstein e Salesin (1995). No presente trabalho, o espaço RGB é usado nos algoritmos de detecção de corte de tomada, já o espaço YIQ é usado na composição de sumário e nas buscas de quadros dos vídeos.

### 3.2.2 Histogramas de cor

Histogramas de cor são uma estimativa da distribuição de cor na imagem. Esta ferramenta é amplamente utilizada para recuperação baseada em conteúdo visual. O histograma pode ser particionado em intervalos de faixas de cores, para cada faixa ou partição, os *pixels* de cor dentro dessa faixa são contados, resultando numa representação das freqüências de ocorrência de cor. Uma vez obtidos os histogramas de diversas imagens, estes podem ser comparados por diversas medidas de distância. Esse é um dos métodos mais simples de serem implementados, mas ainda assim mostra resultados bastante relevantes (DESELAERS; KEYSERS; NEY, 2004).

### 3.2.3 Momentos de cor

Os momentos de cor têm sido usados em muitos sistemas de recuperação de imagens como o QBIC (FLICKNER *et al.*, 1997; FALOUTSOS *et al.,* 1994). Outro ponto a favor do uso dos momentos é que eles carregam uma representação muito compacta da imagem quando comparada com outras características de cor (FENG; SIU; ZHANG, 2003). Gonzalez e Woods (2000) utilizam esses momentos como descritores estatísticos de textura de uma região da imagem em níveis de cinza, ou seja, como característica local.

A partir do histograma associado a cada componente de cor de uma imagem matricial, é possível extrair-se $n$ momentos. O primeiro momento, dado pela eq. 4, é o valor médio da intensidade de cor dos *pixels* da imagem. O segundo momento representa a variância dessa intensidade de cor. O terceiro momento é uma medida de anti-simetria do histograma, enquanto que o quarto é uma medida de sua planaridade. Os demais momentos não são facilmente relacionados com o formato do histograma, mas eles adicionam informação para a caracterização de textura (GONZALEZ; WOODS, 2000). Os momentos em torno da média, a partir do segundo momento são obtidos conforme eq 5.

De acordo com Gonzalez e Woods (2000), seja $z$ uma variável aleatória denotando a intensidade discreta de uma imagem e seja $p(z_i), i=1,2,3,...L$ o histograma correspondente. $L$ representa o número de níveis distintos de intensidade na imagem. O *n*-ésimo momento de $z$ em torno da média é dado pela eq. 5. Note que $\mu_0 = 1$ e $\mu_1 = 0$.

$$m = \sum_{i=1}^{L} z_i p(z_i) \qquad \text{eq. 4}$$

$$\mu_n(z) = \left( \sum_{i=1}^{L} (z_i - m)^n p(z_i) \right) \qquad \text{eq. 5}$$

### 3.2.4 Entropia

Outra característica que pode ser extraída dos quadros de vídeo é a entropia (EPSTEIN, 1988). A entropia representa a quantidade média de informação contida em uma imagem. Outra forma de se abstrarir a entropia é pensar na complexidade da imagem. Quando poucas cores distintas são encontradas na imagem, ela pode ser considerada simples. No caso mais extremo, se uma imagem possui apenas uma cor, ela apresentará a mais baixa quantidade média de informação. À medida que objetos são adicionados à imagem, mais cores precisam ser incluídas, aumentando a quantidade de informação contida nessa mesma imagem. A medida da entropia é mostrada na eq. 6.

$$H = -\sum_{i=1}^{n} p_i \cdot \log_2(p_i) \qquad \text{eq. 6}$$

## 3.3 SISTEMAS DE RECUPERAÇÃO DE VÍDEO BASEADO EM CONTEÚDO VISUAL

Em resposta ao grande crescimento da quantidade de imagens e vídeos produzidos diariamente, foram criadas diversas ferramentas de *CBIR* para recuperação de imagens e de *CBVR* para recuperação de conteúdo visual em vídeos. Vários trabalhos têm sido dedicados à solução do problema da recuperação de vídeos e imagens com base em seus conteúdos. Alguns desses trabalhos, que serviram de base para algumas das idéias aplicadas nessa dissertação, são apresentados e discutidos nas seções a seguir.

### 3.3.1 Query By Image Content

Um dos trabalhos mais conhecidos na área de recuperação de imagem baseada em conteúdo é o *Query By Image Content* (QBIC) desenvolvido pela IBM (FLICKNER *et al*., 1997) nos anos 90. O QBIC permite aos usuários buscar uma imagem usando características computáveis do vídeo tais como: cor, textura, forma, movimentação de câmera e objetos em vídeo, dentre outras informações gráficas. Primeiro os vídeos são segmentados em tomadas, em seguida, para cada tomada extraem-se quadros-chave. Esses quadros extraídos das tomadas são tratados como imagens estáticas. Dessas imagens são extraídas características estatísticas, que depois são armazenadas numa base de dados. Outro processamento realizado sobre as tomadas é a estimativa de movimento dos objetos. Para realizar uma busca, um usuário pode, por exemplo, desenhar o rascunho de uma imagem que tenha um fundo verde com um objeto vermelho centrado, além de poder selecionar numa paleta de texturas, aquelas que mais se assemelham com o padrão que ele procura. As buscas são baseadas em similaridade vetorial, utilizando-se vetores que representam as características das imagens (cor, textura, etc).

## 3.4 BLOBWORLD

O *BlobWorld* (CARSON *et al*., 2002) faz a indexação das imagens a partir dos objetos automaticamente segmentados que as compõem. Para isso a ferramenta transforma uma imagem, que é representada com muitos *pixels*, num pequeno conjunto de regiões de imagens ou segmentos que são coerentes em cor e textura.

Para realizar uma pesquisa, o usuário seleciona um objeto na imagem de busca, e o sistema retorna imagens com objetos semelhantes. Como a ferramenta utiliza como principal característica a segmentação de imagem, que é um problema tipicamente difícil do processamento de imagens (GONZALEZ; WOODS, 2000), alguns grupos de imagens podem ficar super segmentados, dificultado o sucesso das buscas. Para contornar esse problema, uma característica destacada pelos autores é que o sistema permite a visualização da segmentação tanto da imagem de busca quanto das imagens retornadas, permitindo assim uma intervenção manual na segmentação, proporcionando um ajuste mais fino das métricas de similaridade adotadas.

## 3.5 NETRA

O *NeTra* (MA; MANJUNATH, 1999) é um sistema de recuperação de imagens que usa cor, textura e a forma dos objetos. Para o caso da forma, as bordas dos objetos são detectadas na imagem de busca *Q*, que são posteriormente combinadas em contornos fechados. Descritores de forma são computados usando as amplitudes da transformada de Fourier de três tipos de funções de fronteira: curvatura, distância do centróide e funções de coordenada complexas. A similaridade entre imagens é calculada através da distância Euclidiana entre 2 desses descritores.

## 3.6 VISUALSEEK

O *VisualSeek* (SMITH; CHANG, 1997) realiza a recuperação de imagens baseado apenas em características de cor. O diferencial da abordagem é o fato de levar em conta a posição espacial das regiões de cores. Com isso, as buscas consideram não só a posição absoluta como também a posição relativa da ocorrência de regiões de cores.

## 3.7 FAST MULTIRESOLUTION IMAGE QUERY

Além do QBIC, outro trabalho relevante na área de *CBIR* é o *Fast Multiresolution Image Query* (JACOBS; FINKELSTEIN; SALESIN, 1995). No artigo, os autores extraem os maiores coeficientes da transformada *wavelet* da imagem. Esses coeficientes são utilizados na busca, uma vez que, segundo os autores, são os componentes de maior energia e representam a “essência da imagem”. O

trabalho, proposto por Jacobs, Finkelstein e Salesin (1995), compõe parte importante das implementações realizadas na presente pesquisa.

Os trabalhos descritos entre os itens 3.3.1 e 3.7 tiveram um papel fundamental no início desta pesquisa, ajudando a direcionar as linhas e abordagens mais adequadas a serem seguidas para resolução dos problemas propostos. Estes trabalhos também ajudaram a catalogar quais técnicas ou metodologias deixar de lado de acordo com os resultados expostos em suas publicações. Embora estes trabalhos em sua maioria, estejam relacionados com a recuperação de conteúdo visual em imagem, ao invés de vídeo especificamente (exceto o QBIC), eles inspiraram várias fases das implementações realizadas. Características consideradas importantes tais como as encontradas no QBIC e no *Fast Multiresolution Image Query* foram incorporadas nas implementações práticas do ambiente descrito no capítulo 5. Já outras abordagens como o BlobWorld, o VisualSeek e o NeTra, por empregarem segmentação de imagem e/ou métodos considerados mais complexos sem uma relação de benefício atraente, foram deixadas de lado.

## 3.8 MÉTRICAS DE COMPARAÇÃO DE IMAGEM OU QUADROS DE VÍDEO

As características visuais mais simples a serem computadas são as baseadas na informação dos *pixels* que compõem a imagem e nas transformações feitas sobre os mesmos. Podem ser gerados outros domínios, agrupando os *pixels* para extrair diversas características e/ou formar novas entidades para análise. A recuperação de imagens com base em conteúdo visual visa responder questões como as que foram propostas por Gupta e Jain (2007). Um exemplo de questão desse tipo seria: como encontrar todas as imagens que tenham “aproximadamente a mesma cor” na “região central” a partir de uma imagem particular fornecida como exemplo? A “região central” carrega informação espacial na busca, enquanto que a expressão “aproximadamente a mesma cor” deve ser entendida como uma informação de que a diferença entre as cores esteja dentro de um certo limite tolerável. Outro exemplo poderia ser: como encontrar todas as imagens que se repetem com um deslocamento espacial $D$ num certo limite?

Uma abordagem simples para busca de imagens num repositório de dados é apresentada a seguir com o uso das normas $L^1$ ou $L^2$.

Considerando a pesquisa de um quadro alvo *T*, a partir de uma imagem de busca *Q*. Uma das primeiras abordagens a serem usadas é a métrica da norma $L^1$ ou $L^2$ conforme descrito nas eq. 7 e eq. 8, respectivamente.

$$\|Q,T\|_1 = \sum_{i,j} |Q[i,j] - T[i,j]| \qquad \text{eq. 7}$$

$$\|Q,T\|_2 = \sqrt{\sum_{i,j} (Q[i,j] - T[i,j])^2} \qquad \text{eq. 8}$$

O principal problema das normas $L^1$ e $L^2$ para a busca de quadros é o custo computacional. Uma operação de busca obriga a comparação de todos os *pixels* das imagens envolvidas no conjunto cada vez que a busca for solicitada. No caso de um único vídeo, deve-se levar em conta uma quantidade considerável de quadros, tornando inviável a aplicação em bases de dados que excedam um certo limite. Outro problema é a possível incompatibilidade de resolução para comparação entre as imagens *Q* e *T*. Embora seja possível contornar esse problema com relativa facilidade, equalizar diferentes dimensões das imagens aumenta ainda mais o custo computacional da abordagem além de introduzir mais erro (JACOBS; FINKELSTEIN; SALESIN, 1995).

## 3.9 COMPARAÇÃO DAS CARACTERÍSTICAS VISUAIS

Uma abordagem diferente da apresentada na seção 3.8 para a busca de imagens, pode ser feita reduzindo-se a imagem a um vetor *n*-dimensional de características, tais como as descritas na seção 3.2. Com essa abordagem, a busca de uma imagem passa a ser feita sobre estas características representadas por esse vetor, que substitui a imagem no processo de busca. Uma vez definidas quais características irão compor a representação da imagem, elas podem ser abstraídas e imaginadas como sendo um ponto no espaço *n*-dimensional. Desse modo, comparar imagens através desse paradigma passa a ser uma comparação de distâncias entre os pontos que representam as imagens no espaço ou hiper-espaço. Uma variedade de distâncias podem ser usadas, tais como a distância euclidiana (que é a norma $L^2$, e constitui-se na métrica mais usada em sistemas de recuperação de imagens (FENG; SIU; ZHANG, 2003), a distância de *Manhattam*, ou distância no tabuleiro de xadrez, dentre outras (GONZALEZ; WOODS, 2000).

Uma vez que o resultado da busca não será, em geral, uma única imagem, mas sim, um conjunto, classificado de acordo com a medida de similaridade com a imagem de busca *Q*, sempre é necessário considerar um custo computacional alto, que está relacionado, tanto à técnica de comparação, quanto ao tamanho do conjunto pesquisado. Muitas formas de mensurar similaridade foram desenvolvidas para recuperação de imagens baseadas em estimativas empíricas nos últimos anos (DESELAERS, 2003; DESELAERS; KEYSERS; NEY, 2004; FALOUTSOS *et al*., 1994; MA; MANJUNATH, 1999; SMITH; CHANG, 1997). Se as variáveis aleatórias que representam as características da imagem são independentes entre si, e possuem igual importância na representação da imagem, então, pode-se utilizar a distância euclidiana (já mencionada nesse trabalho para outras comparações, como para a própria distância bruta entre *pixels* da imagem descrita na eq. 8) como métrica de similaridade entre quadros. Esta é exatamente uma das abordagens utilizadas neste trabalho para a comparação dos vetores estatísticos representanto o conteúdo de um vídeo. Embora não tenha sido levantado no presente trabalho a independência das variáveis, assume-se por simplificação que as mesmas são independentes entre si. Os detalhes dessa abordagem serão apresentados no item 5.2.

Embora exista uma grande variedade de medidas de distância e comparação de vetores na literatura, o presente trabalho comprara características visuais ou através de distância euclidiana, ou através de discrepância entre assinaturas de imagem, proposta por Jacobs, Finkelstein e Salesin (1995), conforme descrita no item 4.8.2. A comparação de imagem *pixel*-a-*pixel*, descrita nesse capítulo, não é implementada para busca de quadros-chave por ser computacionalmente inviável. Entretanto esse tipo de comparação é realizado entre quadros no processamento do vídeo para ajudar na seleção de quadros-chave, como descrito em mais detalhes nos itens 5.1.1.4 e 6.1.4.

# 4 EXTRAÇÃO DE ASSINATURA DE QUADROS DE VÍDEO BASEADA EM *WAVELET*

As *wavelets* são ferramentas matemáticas usadas para a decomposição hierárquica de funções. Elas permitem que uma função seja descrita ou decomposta como uma parte global adicionada a uma série de detalhes, que vão dos mais amplos aos mais minuciosos. Independente do que representa a função, se é um sinal de áudio, uma imagem ou uma superfície, as *wavelets* provêem uma forma elegante de análise para esses sinais (STOLLNITZ; DEROSE; SALESIN, 1995). Esta análise é comparada a um "microscópio matemático", uma vez que, através dessa técnica, é possível se ter acesso ao sinal em diferentes níveis de detalhamento.

Apesar da primeira menção às *wavelets* ter acontecido em 1909, por Alfred Haar, só recentemente, em 1985, através de um trabalho em processamento digital de imagens, Stephane Mallat trouxe notoriedade ao uso das *wavelets* com a construção da primeira *wavelet* não trivial suave. Com base no trabalho de Mallat, a matemática Ingrid Daubechies criou um conjunto de bases ortogonais de *wavelet* com suporte compacto, já que o suporte não era oferecido pela base de Mallat. Os trabalhos de Daubechies são os alicerces das aplicações atuais com *wavelets* (LIMA 2002).

As aplicações da análise de *wavelet* são, hoje em dia, realmente muito amplas. Na biologia é usada para o reconhecimento de membrana celular, na distinção entre membranas normais e patológicas; na metalurgia é usada para caracterizar superfícies ásperas; nas finanças para detectar rápidas variações de valores; na Internet para descrever o seu tráfego de dados (MISITI *et al*. 2002). Em processamento de imagens, a transformada de *wavelet* pode ser usada para filtrar ruído, compactar dados, e no presente trabalho, para extração de características que são suporte à busca de quadros de vídeo.

Assim como na análise de Fourier, a representação por *wavelet* provê acesso a uma grande variedade de dados em diversos níveis de detalhe. Entretanto, a análise de *wavelet* difere da análise de Fourier no sentido em que a informação da posição local é conservada, enquanto que na transformada de Fourier essas informações são perdidas (WEN; HUFFIMIRE; FINKELSTEIN, 1998). Os aspectos

que levam à escolha da transformada *wavelet* para extração de características no presente trabalho são mostrados mais adiante.

## 4.1 A ANÁLISE DE FOURIER

A análise de sinais conta hoje com uma grande variedade de ferramentas. Provavelmente, a ferramenta mais conhecida seja a Análise de Fourier, que decompõem uma função em suas constituintes senoidais de diferentes freqüências. O que a transformada de Fourier faz é uma mudança no domínio do sinal. No caso específico, uma mudança do domínio com base no tempo para o domínio com base na freqüência. A Figura 6 ilustra essa mudança de paradigma (MISITI *et al*. 2002).


Figura 6 – Transformada de Fourier

Fonte: Misiti *et al*. (2002).

Um sinal é dito estacionário quando suas propriedades estatísticas não mudam ao longo do tempo. A Transformada de Fourier é muito útil em uma grande variedade de aplicações, especialmente quando o sinal tem essa característica estacionária. Entretanto, para sinais não estacionários, há uma desvantagem na transformada de Fourier com relação à informação temporal, pois esta se perde. Isto é, não é possível dizer quando um determinado evento ocorreu no domínio transformado. No caso de funções bidimensionais, como no presente trabalho, os quadros de um vídeo, a informação espacial seria perdida no domínio transformado de Fourier. Essa informação é considerada importante, pois na busca de imagens, quando se desenha um rascunho com um objeto centrado, não se deseja recuperar imagens com o objeto no canto superior esquerdo, por exemplo.

## 4.2 A ANÁLISE DE FOURIER EM JANELA

Como esforço para amenizar a perda de informação tempotal/espacial, Dennis Gabor (1986) adaptou a Transformada de Fourier para analisar segmentos limitados do sinal de forma independente. Uma técnica chamada de "janelamento". A adaptação de Gabor é chamada de *Short-Time Fourier Transform* (STFT), e mapeia

o sinal em uma função bidimensional de tempo e freqüência. A Figura 7 ilustra a idéia proposta por Gabor (MISITI *et al*. 2002).

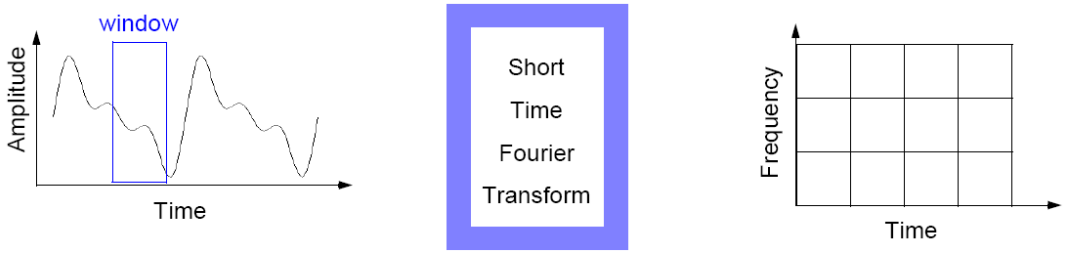


Figura 7 – Análise de Fourier em janela.

Fonte: Misiti *et al*. (2002).

A STFT representa um bom compromisso entre a conservação das informações de tempo e de freqüência. Contudo, sua precisão é limitada e determinada pelo tamanho da janela. Como o tamanho da janela é fixo, alguns trechos do sinal podem não comportar um ciclo senoidal completo, se a janela for pequena, perdendo assim informações de sinais de baixa freqüência. Mesmo janelas maiores podem não vão comportar ciclos de freqüências muito baixos. Em contrapartida, janelas demasiadamente grandes perdem precisão temporal. Em funções bidimensionais, no presente caso, os quadros do vídeo, janelas pequenas podem não comportar objetos grandes, enquanto que janelas muito grandes levam à perda da precisão espacial dos objetos contidos na imagem analisada.

## 4.3 A ANÁLISE *WAVELET*

A Análise de *wavelet*, também chamada de transformada *wavelet* ou ainda, decomposição *wavelet*, representa o próximo passo lógico: uma técnica de "janelamento", onde as janelas têm tamanhos variáveis. A análise *wavelet* usa janelas grandes nas mais baixas freqüências e janelas menores nas altas freqüências de modo a adaptar-se a quaquer frequência. A Figura 8 ilustra uma representação na variação do tamanho das janelas com a análise w*avelet*.


Figura 8 – Transformada *wavelet*.

Fonte: Misiti *et al*. (2002).

A Figura 9 mostra ilustra a representação gráfica dos quatro domínios mencionados anteriormente.

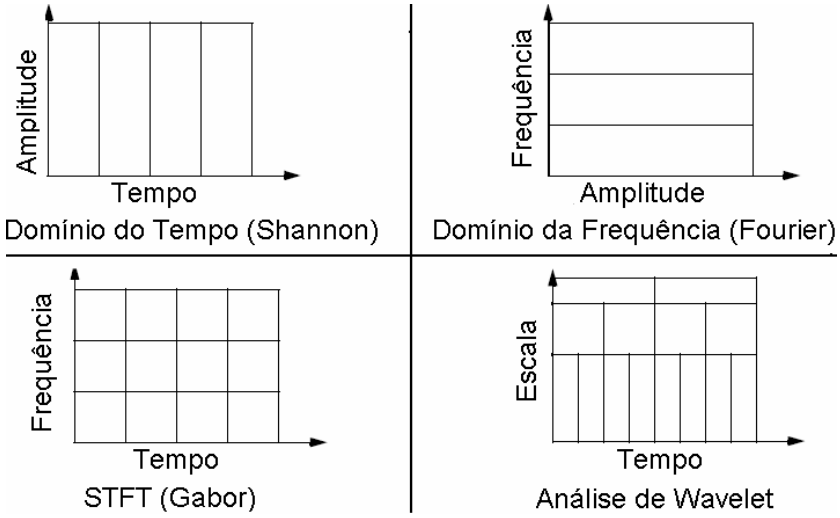


Figura 9 – Representação dos quadro domínios de sinal.

Fonte: Misiti *et al*. (2002).

Sempre que é usada a análise *wavelet,* fala-se em escala e não em freqüência, contudo existe uma relação entre as duas entidades.

Já foram comentadas as vantagens da análise *wavelet*, mas o que é uma *Wavelet*? De acordo com Misiti *et al*. (2002), uma *wavelet* é uma função de duração efetivamente limitada que tem média igual a zero. O nome *wavelet* vem justamente dessa duração limitada que torna a variação da função limitada a um pequeno intervalo, com isso uma possível tradução para *wavelet* pode ser ondinha ou ondeleta.

Comparando *wavelets* com ondas senoidais, que são a base da análise de Fourier, na Figura 10, observa-se que as ondas senoidais não têm uma duração limitada, prolongando-se entre mais e menos infinito, diferentemente da *wavelet*. Enquanto a função seno é suave e previsível, a wavelet é assimétrica e irregular.

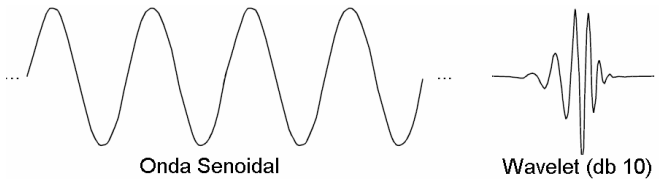


Figura 10 – Função Seno e *wavelet* (db10).

Fonte: Misiti *et al*. (2002).

Assim como na análise de Fourier um sinal é decomposto em funções senoidais de diversas freqüências, na análise *wavelet*, um sinal é decomposto nas funções derivadas da *wavelet* mãe em diversas escalas e deslocamentos temporais.

## 4.4 WAVELET MÃE

A *wavelet* mãe é uma função que tem média zero e decai bruscamente de maneira oscilatória. A partir dessa função base, são geradas funções filhas, representadas via superposição de versões dilatadas e transladadas da *wavelet* mãe pré-especificada.

Uma série de *wavelets* mãe foram definidas. Dentre as principais destacam-se: a *wavelet* de *Haar*; a família *Daubechies*, `dbN`; *Biorthogonal*; *Coiflets*; *Symlets*; *Morlet*; Chapéu Mexicano e *Meyer*.

Existe uma variedade de *wavelets* mãe, a escolha de qual delas é melhor depende da aplicação e do tipo do sinal a ser analisado.

### 4.4.1 Haar

Qualquer discussão a respeito das *wavelets* começa com a primeira e mais simples: Haar. A mesma é descontínua e remonta uma função degrau. A *wavelet* de *Haar* é igual a *Daudechies* `db1`. A Figura 11 mostra a sua forma.

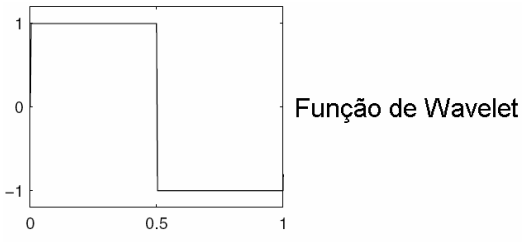


Figura 11 – *Wavelet* de Haar

Fonte: Misiti *et al.* (2002).

A definição da função de *wavelet* mãe, $\psi$, de Haar e da sua função de escala $\phi$ são mostradas a seguir:

$$\psi(x) = 1, \quad \text{se} \quad x \in [0,0.5[$$
$$\psi(x) = -1, \quad \text{se} \quad x \in [0.5,1[$$
$$\psi(x) = 0, \quad \text{se} \quad x \notin [0,1[$$

$$\phi(x) = 1, \quad \text{se} \quad x \in [0,1]$$
$$\phi(x) = 0, \quad \text{se} \quad x \notin [0,1]$$

### 4.4.2 Família Daubechies

Ingrid Daubechies, uma das mais importantes pesquisadoras de *wavelets*, criou uma família de ondeletas ortogonais com suporte compacto que permitem a DWT (transformada discreta de *wavelet*) computacionalmente praticável.

O nome dessa família de funções é mesmo nome da sua criadora, Daubechies ou `dbN`, onde `N` é a ordem. Como mencionado anteriormente, a `db1` é a própria *wavelet* de *Haar*. A Figura 12 mostra as funções *psi* entre a `db2` e `db7` (MISITI *et al*. 2002).

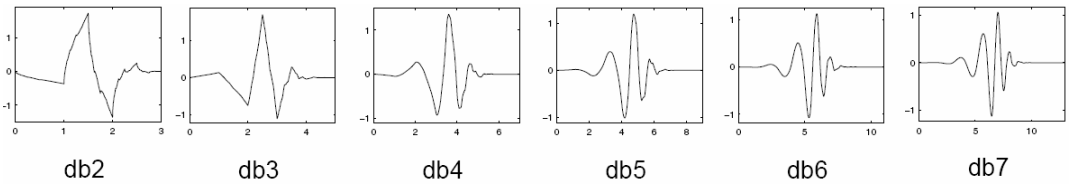


Figura 12 – Família de *Wavelet Daubechies* dbN.

Fonte: Misiti *et al*. (2002).

Para detalhes a respeito das funções de *wavelet* $\psi$, e escala $\phi$, da família `dbN` e outras categorias de funções de *wavelet*, vide (MISITI *et al*. 2002).

## 4.5 A TRANSFORMADA *WAVELET* DE CONTÍNUA

Antes de mostrar a transformada contínua de *wavelet*, apresenta-se uma breve revisão a respeito da transformada contínua de Fourier.

Matematicamente, o processo da análise de Fourier é representado pela transformada de Fourier conforme eq 9.

$$F(\omega) = \int_{-\infty}^{+\infty} f(t) e^{j\omega t} dt \qquad \text{eq. 9}$$

Esta equação representa a soma infinitesimal ao longo de todo tempo da função $f(t)$ multiplicado por uma exponencial complexa. Vale lembrar que uma exponencial complexa pode ser dividida em uma componente cossenoidal real, e uma senoidal complexa.

O resultado da transformada de Fourier são os coeficientes $F(\omega)$, que representam componentes constituintes do sinal. Graficamente, o processo se parece com a Figura 13.

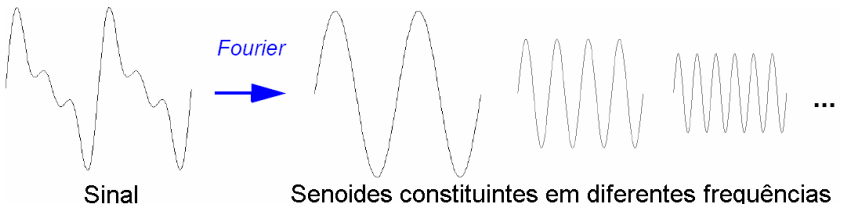


Figura 13 – Representação da Transformada Contínua de Fourier.

Fonte: Misiti *et al*. (2002).

De modo similar à transformada de Fourier, a transformada contínua *wavelet* ou CWT (*Continuous Wavelet Transform*) é definida como a integral no tempo do sinal multiplicado pelas diferentes versões da *wavelet* mãe, $\psi$, em infinitas escalas, conforme representado na eq. 10.

$$\mathrm{C}(\text{escala}, \text{posição}) = \int_{-\infty}^{+\infty} f(t)\psi(escala, posição, t)dt \qquad \text{eq. 10}$$

Os resultados da CWT são infinitos coeficientes de wavelet *C*, em função da escala e posição.

De modo similar à Fourier, multiplicando-se cada coeficiente obtido pela escala e deslocamento apropriado da função de *wavelet*, produzem-se as *wavelets* constituintes do sinal original, como ilustrado na Figura 14.

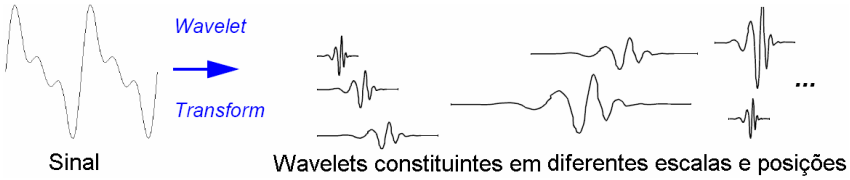


Figura 14 – Representação da Transformada Contínua de *Wavelet*.

Fonte: Misiti *et al*. (2002).

### 4.5.1 Escala

A análise de *wavelet* produz uma visão em tempo-escala do sinal. Nesta seção, será mostrado o efeito da escala da *wavelet*. Escalar uma *wavelet* é, em outras palavras, dilata-la ou comprimi-la.

A letra *a* denota o fator de escala das funções de *wavelet*, que é inversamente proporcional à freqüência analisada. Considerando-se a função senoidal, a mudança de escala é ilustrada na Figura 15.

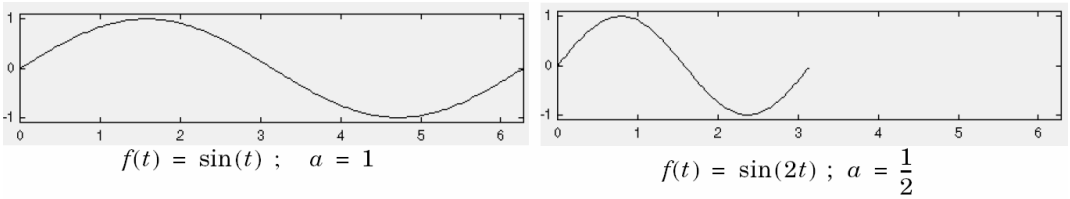


Figura 15 – Fator de escala para função senoidal.

Fonte: Misiti *et al*. (2002).

O fator de escala trabalha exatamente da mesma forma com as *wavelets*. Quanto menor o fator de escala, mais "comprimida" fica a *wavelet*. A Figura 16 ilustra o efeito da mudança de escala aplicado a uma função de *wavelet*.

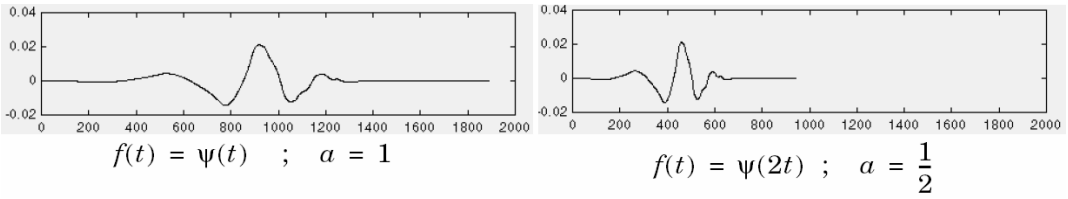


Figura 16 – Fator de escala para *wavelets*.

Fonte: Misiti *et al*. (2002).

Quanto menor a escala *a*, mais comprimida será a *wavelet*, o que torna mais similar (potencialemnte) aos detalhes e rápidas variações no sinal analisado. As menores escalas relacionam-se com as mais altas freqüências e vice-versa. Desse modo, observa-se que existe uma correspondência entre a escala da transformada *wavelet* e a freqüência na transformada de Fourier. A relação matemática formal entre as entidades de escala e freqüência pode ser encontrada em (MISITI *et al*. 2002).

### 4.5.2 Deslocamento

Deslocar uma *wavelet* é simplesmente atrasar ou defasar a onda. Matematicamente, o atraso da função $f(t)$ por $k$ é representado por $f(t-k)$. A Figura 17 ilustra o efeito do processo de deslocamento da *wavelet*.

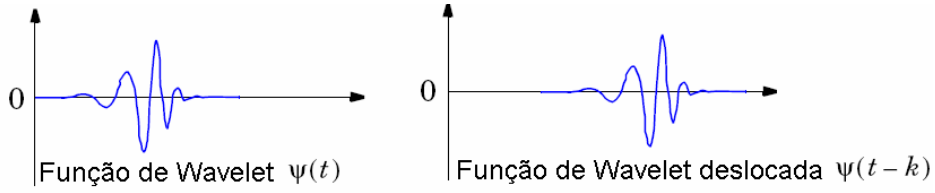


Figura 17 – Efeito de deslocamento da wavelet.

Fonte: Misiti *et al*. (2002).

### 4.5.3 A transformada contínua de *Wavelet* em cinco passos

A transformada contínua de *wavelet* é a soma, infinitesimal (ou integral) sobre todo o tempo do sinal multiplicado pelas diferentes versões da *wavelet* em escala e deslocamento. Esse processo produz os coeficientes da *wavelet* que são funções de escala e posição.

A seguir, são mostrados cinco passos para a transformada contínua de *wavelet*:

1 – Uma *wavelet*, com deslocamento e escala específicas é comparada ao segmento inicial do sinal.

2 – Para a comparação, calcula-se um coeficiente *C*, que representa quão correlacionado está a *wavelet* com a seção interceptada do sinal. Quanto maior o valor de *C*, maior a similaridade entre o segmento do sinal analisado e aquela *wavelet*.

Nota: é importante frisar que os resultados vão depender do tipo da *wavelet* escolhida. A Figura 18 ilustra a comparação de um segmento de sinal com uma *wavelet*.

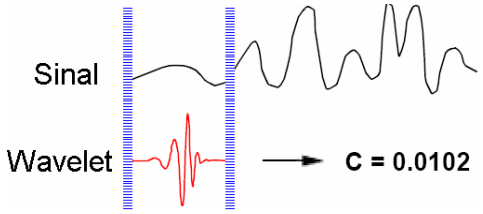


Figura 18 - Comparação entre uma *wavelet* e um segmento do sinal.

Fonte: Misiti *et al.* (2002).

3 – Desloca-se a *wavelet* para a direita, e repetem-se os passos 1 e 2 até que todo sinal seja todo coberto. A Figura 19 ilustra esse efeito de descolamento.

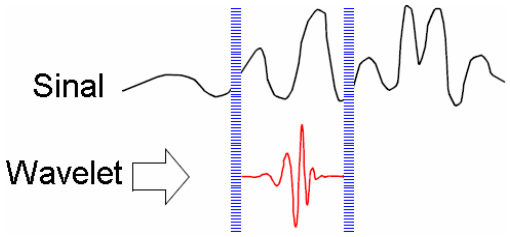


Figura 19 – Descolamento da *wavelet* em um segmento do sinal.

Fonte: Misiti *et al.* (2002).

4 – Aplica-se o fator de escala na *wavelet* e repetem-se os passos de 1 a 3 com essa nova escala. A Figura 20 ilustra o processo de escala da *wavelet* sobre um segmento do sinal.

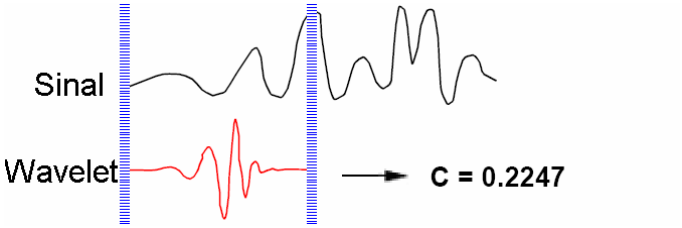


Figura 20 – Efeito de escala na comparação do sinal.

Fonte: Misiti *et al*. (2002).

5 – Repetem-se os passos de 1 a 4 para todas as escalas.

Uma vez concluído o processo, será obtida uma série de coeficientes de *wavelet* em tempo-escala. Uma forma de visualizar esses coeficientes é como uma função de intensidade, como na Figura 21. O tempo e a escala são representados como coordenadas espaciais e o valor do coeficiente é representado pela luminosidade do ponto (tempo-escala) no espaço, sendo os valores mais escuros, os coeficientes menores e os pontos mais claros, os coeficientes maiores indicando grande semelhança entre sinal e *wavelets* respectivas.

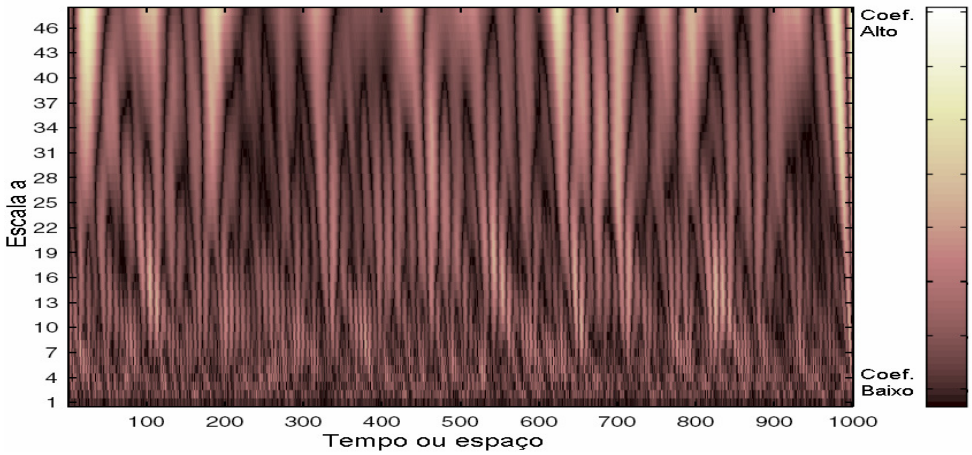


Figura 21 – Coeficientes da CWT mostrados com função de intensidade.

Fonte: Misiti *et al*. (2002).

É importante notar que o fato da análise *wavelet* não produzir informação em tempo-freqüência não é uma deficiência, mas na verdade, uma grande vantagem da técnica. O formato tempo-escala não é apenas uma forma diferente de visualizar os dados, é também uma forma mais natural para muitos fenômenos. Para detalhes matemáticos a respeito da transformada de *wavelet*, é recomendada a leitura de

(CHUI, 1992; DAUBECHIES, 1992; A DEVORE; JAWERTH; LUCIER, 1992; STOLLNITZ; DEROSE; SALESIN, 1995).

## 4.6 A TRANSFORMADA DISCRETA DE WAVELET

Em sua definição matemática formal, a transformada contínua *wavelet* é aplicada a um sinal com resolução temporal infinita e, por conseguinte, precisa de infinitas escalas e deslocamentos temporais infinitamente suaves gerando assim infinitos coeficientes.

Em contrapartida, a proposta da transformada discreta *wavelet* (DWT) é escolher um subconjunto de escalas e deslocamentos onde serão feitos os cálculos. Nessa técnica, são usadas escalas e posições baseadas em potência de dois. Essas escalas e posições são chamadas respectivamente de *dyadic scale* e *dyadic position*. A DWT é muito mais rápida computacionalmente e economiza memória.

### 4.6.1 Decomposição por filtragem

Em muitos sinais, as baixas freqüências contém a parte mais importante do sinal, carregando suas principais características, pode-se dizer que é o caso da maioria das imagens. Os componentes de alta freqüência por outro lado, carregam informação, que enriquecem o sinal em detalhes. A voz humana é um exemplo. Se removido os componentes de mais alta freqüência, a percepção do som soa diferente, contudo, as palavras continuam sendo entendidas. No caso das imagens, se removidos os componentes de alta freqüência, os objetos parecem um pouco borrados e as bordas se espalham, mas mesmo assim, a imagem continua sendo entendida em seu contexto semântico. Os componentes de alta freqüência apenas enriquecem a imagem, deixando-a mais detalhada.

Na análise *wavelet*, é comum se falar em aproximação e detalhes. A aproximação está ligada aos componentes de alta-escala, ou seja, de baixa freqüência do sinal. Os detalhes estão ligados à baixa-escala, ou seja, aos componentes de alta freqüência.

Na DWT, utiliza-se um conceito de decomposição por filtragem. Resumidamente, o sinal passa por dois filtros, um filtro passa-baixa e outro passa-alta. O processo é repetido recursivamente sobre o resultado anterior do filtro passa baixa, até que não seja mais possível continuar a decomposição, porque o filtro

passa-baixa reduz a quantidade de amostras pela metade. A Figura 22 mostra o esquema de filtragem do sinal em alta e baixa freqüências.

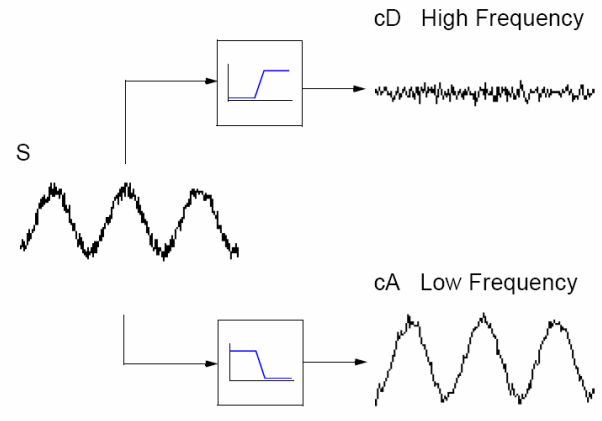


Figura 22 – Filtros Passa Baixa e Passa Alta.

Fonte: Misiti *et al*. (2002).

A decomposição continua recursivamente, e a cada recursão o sinal de baixa freqüência vai decaindo em quantidade de amostras até que não seja mais possível continuar a decomposição. Essa decomposição em vários níveis é chamada de *Wavelet decomposition tree*, ou árvore de decomposição de *wavelet* mostrada na Figura 23. Nesta figura, *S* representa o sinal, $cA_n$ representa o sinal após o filtro passa baixas e $cD_n$ representa o sinal após o filtro passa altas no nível *n*.

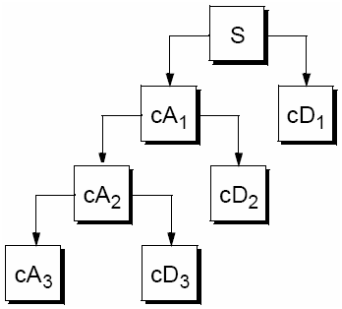


Figura 23 – Árvore de decomposição.

Fonte: Misiti *et al*. (2002).

Os coeficientes de *wavelet* podem passar pelo processo inverso da transformada recompondo o sinal original, a este processo dá-se o nome de transformada inversa de *wavelet*. Recomenda-se a leitura de Misiti, Michel *et al*. (2002) para detalhes a respeito da transformada inversa.

### 4.6.2 DWT pela *wavelet* de Haar

Embora internamente os filtros operem com convolução (MISITI *et al*. 2002), uma operação relativamente cara computacionalmente, a *wavelet* de Haar oferece

algoritmo computacionalmente mais barato que permite que a mesma decomposição seja feita de modo bem mais rápido. O processo será mostrado no decorrer dessa seção.

Para exemplificar essa transformada discreta da *wavelet* de Haar, pode-se observar o seguinte vetor com apenas quatro amostras como exemplo:

[9 7 3 5]

Pode-se decompor esse simples vetor com a transformada Haar, para isso, primeiro calculam-se as médias das amostras em pares adjacentes. Essa operação representa o filtro passa baixa. Ao final será obtido o seguinte vetor:

[8 4]

Obviamente, alguma informação é perdida no processo de calculo das médias. Para recuperar os quatro valores do vetor original, faz-se necessária uma informação adicional, que é obtida pelo coeficiente de detalhe. Nesse exemplo, o coeficiente de detalhe é igual a 1, uma vez que a primeira média que computada é uma unidade menor que 9 e uma unidade maior que 7. Esse coeficiente de detalhe, permite recuperar os dois primeiros valores do vetor original. Similarmente, para a segunda média, o segundo coeficiente de detalhe é $-1$, já que $4+(-1)=3$ e $4-(-1)=5$.

Desse modo, é possível decompor o sinal original em um sinal de baixa freqüência com duas médias, mais um sinal de alta freqüência com dois coeficientes de detalhe, como teria sido feito pelo banco de filtros. Repete-se esse processo recursivamente, agora aplicando ao vetor de médias até que reste apenas uma média geral. Ao final obtém-se toda a transformada *wavelet* do sinal, com uma informação global, que é a média, e as informações de detalhes.

| Resolução | Médias | Coficientes de detalhes |
|---|---|---|
| 4 | $[\ 9 \quad 7 \quad 3 \quad 5\ ]$ | |
| 2 | $[\ 8 \quad 4\ ]$ | $[\ 1 \quad -1\ ]$ |
| 1 | $[\ 6\ ]$ | $[\ 2\ ]$ |

Finalmente, o vetor original com quatro valores é transformado num vetor contento a média global do vetor original, acrescido de três valores de coeficientes de detalhe. Note que o vetor resultante é perfeitamente reversível e têm o mesmo tamanho do vetor original. Nenhuma informação foi adicionada ou perdida. O vetor resultante é:

[6 2 1 -1]

Nesse exemplo, já é possível observar uma característica importante dessa visão do sinal. Funções com alto grau de redundância, como som, imagem e vídeo, quando transformados por wavelet apresentam a maioria dos coeficientes de detalhe com valores próximos de zero. É essa característica que permite aplicar compressão com perda nesse tipo de dado, zerando os valores mais próximos de zero, ou no caso do presente trabalho, escolhendo os maiores coeficientes para caracterizar os componentes mais importantes do snial. Essa técnica de escolha dos maiores coeficientes é melhor detalhada na seção 4.8, onde se obtém a assinatura da imagem ou do quadro.

4.6.2.1Ortogonalidade

Um atributo crucial para a decomposição e reconstrução perfeita do sinal original com a técnica da transformada *wavelet* é a ortogonalidade entre as *walavets* em escalas e posições diferentes. Embora nem todas as bases de *wavelet* possuam essa importante característica, a *wavelet* de Haar a possui. Um base ortogonal é aquela em que todas as funções bases, no caso as funções $\phi$, devem ser ortogonais entre si. Além da ortogonalidade da função $\phi$, todas as *wavelets* de Haar também são ortogonais entre si nas suas diferentes escalas.

4.6.2.2 Normalização

Outra propriedade desejável é a normalidade. Uma função base $u(x)$ está normalizada se $\langle u|u\rangle = 1$. A normalização pode ser realizada após a transformada ou durante a mesma. O Algoritmo 2 traz a decomposição de um vetor com as funções $\phi$, de *wavelet* e a normalização. Para detalhes sobre a obtenção do fator de normalização vide (STOLLNITZ; DEROSE; SALESIN, 1995).

```
procedure DecompositionStep (C: array [1..h] of reals)
   for i = 1 to h/2 do
      C[i] = (C[2i-1] + C[2i])/√2
      C[h/2 + i] = (C[2i-1]-C[2i])/√2
   end for
   C = C'
end procedure

procedure Decomposition (C: array [1..h] of reals)
   C = C/√h  (normalização dos coeficientes)
   while h > 1 do
      DecompositionStep (C[1..h])
      h = h2
```

```
   end while
end procedure
```

Algoritmo 2 - Decomposição discreta de *wavelet*

Fonte: Stollnitz, Derose e Salesin (1995)

## 4.7 A TRANSFORMADA DISCRETA DE *WAVELET* EM 2D

Os fundamentos da transformada *wavelet* bidimensional são basicamente os mesmos dos descritos anteriormente. As funções bidimensionais tratadas nesse trabalho são os quadros de vídeo. Nessa seção será mostrada a extensão da transformada para duas dimensões.

Existem duas formas de decomposição bidimensional: a decomposição padrão e a não padrão. Para obter a decomposição padrão de uma imagem, primeiramente aplica-se a transformada de *wavelet* em cada linha da matriz, assim, o primeiro elemento de cada linha terá a sua respectiva média e todos os seus demais valores serão coeficientes de detalhe. Uma vez completado esse processo, procede-se a transformada de *wavelet* nas colunas da imagem resultante do passo anterior, ou seja, na imagem já transformada em linhas. Ao final, o primeiro valor será a média geral da imagem, todos os demais coeficientes serão os detalhes. O Algoritmo 3 implementa a decomposição padrão.

```
procedure StandardDecomposition (C: array [1..h, 1..w] of reals)
   for row = 1 to h do
      Decomposition (C[row, 1..w])
   end for
   for col = 1 to w do
      Decomposition (C[1..h, col])
   end for
end procedure
```

Algoritmo 3 – Decomposição de imagem pela transformada de wavelet padrão.

Fonte: Stollnitz, Derose e Salesin (1995)

A Figura 24 ilustra o efeito da decomposição padrão.


Figura 24 – Passos da decomposição padrão numa imagem.

Fonte: Stollnitz, Derose e Salesin (1995)

Uma outra forma de decomposição admitida é chamada de decomposição não padrão. Nesse algoritmo, primeiramente, realiza-se um passo do processo de filtragem passa baixa e passa alta em cada linha do quadro. Uma vez concluído esse processo realiza-se um passo da filtragem nas colunas. Para completar a transformação, repetem-se recursivamente os dois primeiros passos no quadrante resultante do filtro passa baixa, até que toda imagem seja decomposta. O Algoritmo 4 mostra a decomposição não padrão.

```
procedure NonstandardDecomposition(C: array [1..h, 1..h] of reals)
   C = C/h (normalização dos coeficientes)
   while h > 1 do
      for row = 1 to h do
         DecompositionStep (C[row, 1..h])
      end for
      for col 1 to h do
         DecompositionStep (C[1..h, col])
      end for
      h = h/2
   end while
end procedure
```

Algoritmo 4 - Decomposição de imagem pela transformada *wavelet* não padrão.

Fonte: Stollnitz, Derose e Salesin (1995)

A Figura 25 ilustra o efeito da decomposição não padrão


Figura 25 - Passos da decomposição não padrão numa imagem.
Fonte: Stollnitz, Derose e Salesin (1995)

### 4.7.1 Aspectos das busca de quadros por assinatura da imagem

A abordagem de busca baseada em estatísticas é recomendada para os casos em que as imagens alvo *T*, são visualmente semelhantes à imagem de exemplo *Q*, independente da posição espacial ou da forma dos objetos contidos em um quadro. Uma técnica diferente de recuperação que leva em consideração a posição e forma dos objetos é baseada na transformada de *wavelet* (CHUI, 1992; DAUBECHIES, 1992; A DEVORE; JAWERTH; LUCIER, 1992; STOLLNITZ, DEROSE; SALESIN, 1995). Os autores Jacobs, Finkelstein e Salesin (1995) propuseram uma forma rápida e efetiva para uma base de imagens convencionais. No caso específico do presente trabalho, o método foi aplicado para quadros de vídeo com o objetivo de encontrar quadros semelhantes, tanto a partir de imagens fornecidas “por exemplo” como “por rascunho”.

A grande vantagem em se usar *wavelet* para recuperação de imagem baseada em conteúdo é que essa técnica permite uma busca rápida computacionalmente e requer pouco espaço de armazenamento para as “assinaturas das imagens”. Esse foi o nome dado por Jacobs, Finkelstein e Salesin (1995) às características mais significativas da imagem pela decomposição de *wavelet*. A decomposição *wavelet* permite uma boa aproximação da imagem com poucos coeficientes. Essa característica já foi bastante explorada na compressão de

imagem (A DEVORE; JAWERTH; LUCIER, 1992), inclusive gerando o padrão JPEG2000 (CHRISTOPOULOS, C. *et al.,* 2000).

Uma imagem, ao ser passada para o domínio das *wavelets*, sofre uma análise em **multiresolução**, assim cada objeto da imagem vai "casar" melhor em alguma escala e produzir um alto coeficiente com a coordenada espacial da ocorrência do mesmo. É possível destacar outras vantagens para a busca baseada em transformada de *wavelet*: a decomposição é computacionalmente rápida requerendo um tempo linear em relação ao tamanho da imagem e sua implementação requer pouco código (JACOBS; FINKELSTEIN; SALESIN, 1995).

## 4.8 O MÉTODO DE EXTRAÇÃO DA ASSINATURA DA IMAGEM

Os autores Jacobs, Finkelstein e Salesin (1995) propuseram uma estratégia para busca de imagens em bases de dados. Esta estratégia foi adaptada nesta pesquisa para trabalhar sobre os quadros (conteúdo) de um vídeo. Nessa estratégia, o usuário expressa a busca através de uma imagem de entrada *Q*, que pode ser um rascunho feito por ele mesmo, ou uma imagem comum, em princípio independente de resolução. É dito "em princípio, independente de resolução", porque embora a transformada de *wavelet* seja dependente da resolução da imagem de entrada, os coeficientes de maoir valor absoluto que são selecionados para compor a assinatura da imagem tendem a ser basicamente os mesmos em diferentes resoluções, isso obviamente se for a mesma imagem em diversas versões de resolução.

Alguns fatores tornam o processo de busca difícil para o algoritmo de busca. A imagem ser pesquisada, feita como rascunho, é tipicamente diferente da imagem alvo *T*, então o método de busca deve permitir algumas distorções como posição, forma e cor dos objetos que compõem a cena da imagem. O grau dessas distorções pode variar bastante de um individuo para outro, o que é um fator completamente fora do controle do algoritmo.

Se a imagem é fornecida como exemplo, alguns problemas, como ruído na aquisição da imagem, que possam provocar algum deslocamento no brilho ou na composição das cores, precisam ser contornados. Por fim, o algoritmo precisa ser rápido o suficiente para que o resultado em grandes bases de vídeos seja ágil para atender ao anseio e as necessidades do usuário.

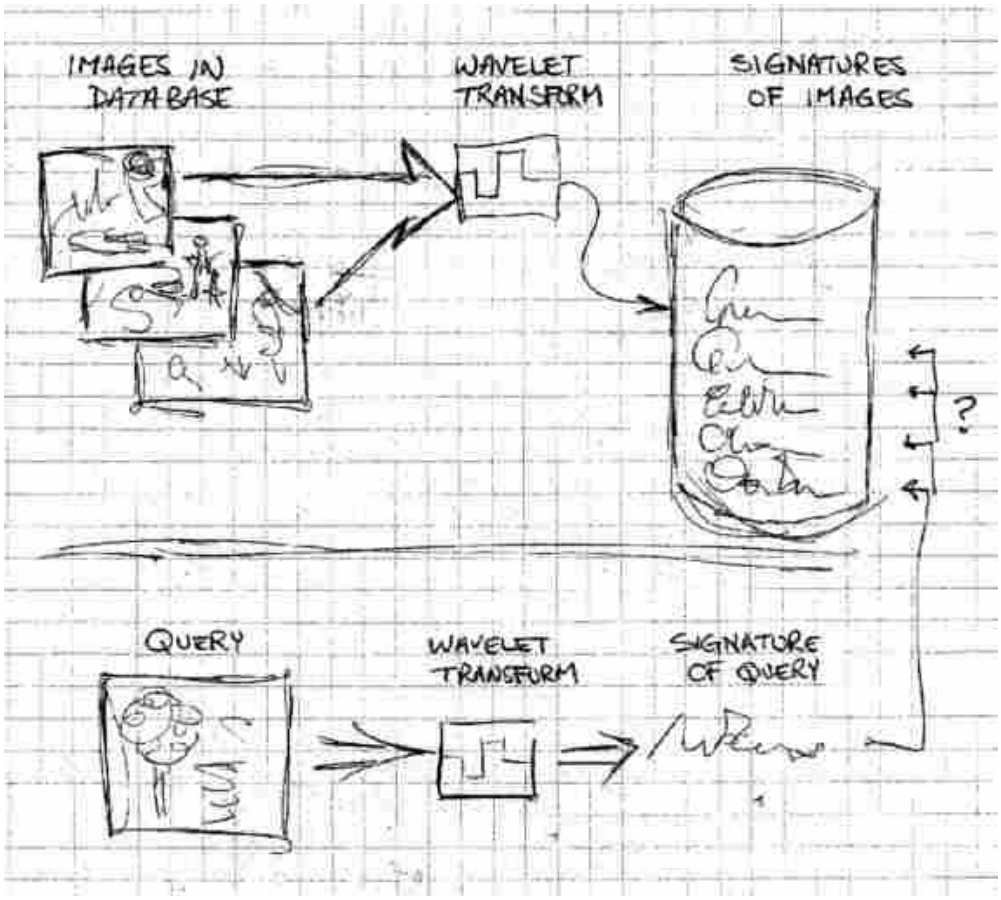


Figura 26 – Modelo de extração e comparação de assinatura de imagens geradas por análise *wavelet*.

Fonte: Jacobs, Finkelstein e Salesin (1995).

Para compor o algoritmo de aquisição da assinatura do quadro ou da imagem, uma série de parâmetros precisam ser definidos. Os parâmetros escolhidos no presente trabalho foram os mesmos do original (JACOBS; FINKELSTEIN; SALESIN, 1995) pelo fato do trabalho já ser bastante consolidado e outros trabalhos já terem repetido também a mesma configuração do algoritmo, como é o caso do recente trabalho de Lui, Rosenberg e Rowley (2007).

### 4.8.1 Parâmetros para a extração da assinatura

O tipo da *wavelet* escolhido foi a de Haar (STOLLNITZ; DEROSE; SALESIN, 1995), pois é de fácil implementação e computacionalmente rápida. Além disso, imagens de rascunho ou *sketch* apresentam superfícies com regiões de cor constante e descontinuidade nas bordas dos objetos.

Para a decomposição em *wavelets* temos duas possibilidades, a decomposição padrão e a não padrão. Embora a decomposição não padrão seja

ligeiramente mais “barata”, do ponto de vista computacional, o trabalho de Jacobs, Finkelstein e Salesin (1995) mostra que a decomposição padrão apresenta melhores resultados em todos os espaços de cor testados (RGB, HSV e YIQ), além de seu código ser um pouco mais simples. Por ser mais eficiente nos resultados das buscas, a decomposição padrão foi escolhida para extração dos coeficientes dos quadros dos vídeos e das imagens de busca.

Antes de decompor um quadro pela transformada *wavelet*, cada quadro capturado é sub-amostrado para a resolução de 128x128 *pixels*. Isso é necessário para satisfazer uma restrição do algoritmo, onde cada dimensão da imagem deve ter $2^n$ *pixels* e também, pelo fato de uma imagem de 128x128 ser de custo computacional mais baixo do que a imagem com resolução original. A imagem em menor resolução perde detalhes, ou componentes de alta freqüência, justamente detalhes que o usuário tende a não desenhar numa imagem de rascunho. Assim, a sub-amostragem atende de maneira adequada à busca por rascunho sem maiores problemas.

De acordo com Jacobs, Finkelstein e Salesin (1995), foram escolhidos, de cada quadro do vídeo processado, 20 coeficientes em cada canal de cor. Ou seja, para cada componente YIQ são selecionados os 10 maiores coeficientes positivos e os 10 menores coeficientes negativos, totalizando assim 60 coeficientes para cada quadro. Esse conjunto de coeficientes carrega a essência da imagem ou como os autores chamaram **“assinatura da imagem”**. A Figura 27 mostra a transformada inversa de *wavelet* com diferentes quantidades de coeficientes ilustrando a idéia de que poucos coeficientes aproximam uma imagem reconstruída com a imagem original.

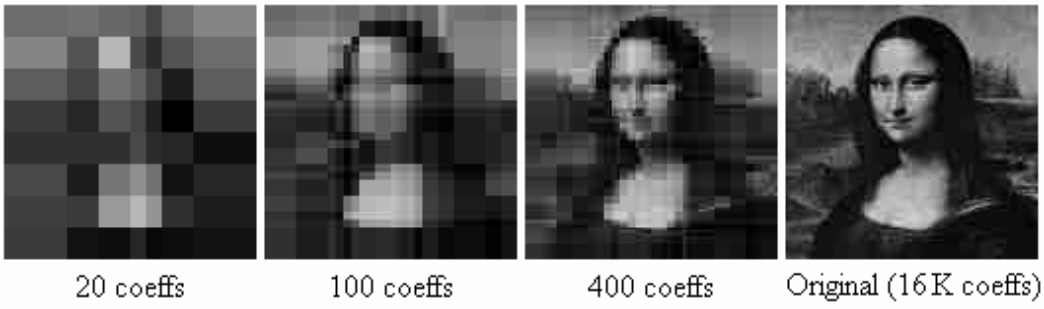


Figura 27 – Imagem reconstruída com diferentes quantidades de coeficientes de *wavelet*.

Fonte: Jacobs, Finkelstein e Salesin (1995).

Nessa abordagem baseada em *wavelets* para a recuperação de imagens, o valor real dos coeficientes que compõem a assinatura da imagem não é tão relevante, pois eles carregam pouca ou nenhuma informação adicional. Assim, os coeficientes mais energéticos são quantizados para -1 ou +1, para os coeficientes negativos e positivos respectivamente. Essa quantização, além de incrementar o poder computacional e reduzir o custo das buscas, ainda parece ter um poder discriminatório maior que o valor real dos coeficientes (JACOBS; FINKELSTEIN; SALESIN, 1995). Apenas os coeficientes de médias nos três canais são armazenados em seu valor real, além desses entram as posições espaciais dos coeficientes de detalhe.

Os mesmos parâmetros usados para a extração da assinatura dos quadros do vídeo devem ser usados para a extração da assinatura da imagem de busca. Qualquer alteração, como o tipo da *wavelet*, da decomposição ou o espaço de cor, torna incompatível a assinatura para comparação dos quadros alvo com as imagens de busca.

### 4.8.2 A métrica de comparação de assinaturas

Nessa seção são mostrados e discutidos os parâmetros para a composição da métrica de busca de quadros.

Antes de ser mostrada a métrica de comparação de assinaturas de imagem, primeiramente fazem-se necessárias algumas definições. Será chamada de *Q* a imagem de busca fornecida como exemplo ou rascunho, e *T* o quadro-chave alvo que se pretende recuperar. Sendo *Q*[0,0] e *T*[0,0] os respectivos coeficientes de média da imagem de busca *Q* e do quadro alvo *T*, e sendo $Q'[i,j]$ e $T'[i,j]$ os coeficientes de detalhe quantizados para -1 e 1 como descrito em 4.8.1, a métrica de comparação de assinatura se dá pela eq. 11.

$$w_0\left|Q[0,0]-T[0,0]\right| - \sum_{i,j:Q'[i,j]\neq 0} w_{bin(i,j)}\left(Q'[i,j]=T'[i,j]\right) \qquad \text{eq. 11}$$

Por conveniência, definem-se os coeficientes $Q'[0,0]$ e $T'[0,0]$, que não correspondem a nenhum coeficiente de detalhe, como sendo iguais a 0. A expressão $a=b$, na equação $Q'[i,j]=T'[i,j]$, vale 1 se verdadeira e 0 se falso.

Nessa métrica de comparação de assinatura proposta por Jacobs, Finkelstein e Salesin (1995), foi criado um conjunto de pesos $w_{bin}$ para a busca a partir de

exemplo e outra de rascunho. A função $bin(i, j)$ provê uma forma de agrupar diferentes coeficientes num pequeno número de *bins*, com cada *bin* pesando uma constante $w_{bin}$. Para um dado conjunto de *bins*, os autores definiram os pesos experimentalmente, e os detalhes desses experimentos encontram-se no trabalho original (JACOBS; FINKELSTEIN; SALESIN, 1995). A função *bin* é definida na eq. 12:

$$bin(i, j) \coloneqq \min\{\max(i, j), 5\} \qquad \text{eq. 12}$$

Para a base de quadros, foi utilizada a mesma tabela de pesos original proposta para o espaço de cor YIQ.

Tabela 1 – Pesos dos coeficientes de *wavelet* para comparação de assinaturas de imagens

| | Rascunho | | | Exemplo | | |
|---|---|---|---|---|---|---|
| *bin* | $w^Y[bin]$ | $w^I[bin]$ | $w^Q[bin]$ | $w^Y[bin]$ | $w^I[bin]$ | $w^Q[bin]$ |
| 0 | 4.04 | 15.14 | 22.62 | 5.00 | 19.21 | 34.37 |
| 1 | 0.78 | 0.92 | 0.40 | 0.83 | 1.26 | 0.36 |
| 2 | 0.46 | 0.53 | 0.63 | 1.01 | 0.44 | 0.45 |
| 3 | 0.42 | 0.26 | 0.25 | 0.52 | 0.53 | 0.14 |
| 4 | 0.41 | 0.14 | 0.15 | 0.47 | 0.28 | 0.18 |
| 5 | 0.32 | 0.07 | 0.38 | 0.30 | 0.14 | 0.27 |

Fonte: Jacobs, Finkelstein e Salesin (1995).

## 4.9 DISCUSSÕES FINAIS

Esse capítulo apresentou uma revisão nos fundamentos da transformada *wavelet* que são aplicados na análise e extração de características dos quadros de vídeo. Essa análise gera um vetor característico que representa o conteúdo visual do quadro de forma bastante compacta. Isso é importante para indexar os quadros analisados e obter uma forma rápida e efetiva na recuperação dos mesmos.

# 5 ESTUDO DE CASO: AMBIENTE DE INDEXAÇÃO E RECUPERAÇÃO DE CONTEÚDO EM VÍDEOS

Este capítulo apresenta a arquitetura do ambiente[5] desenvolvido para indexação e recuperação de quadros de vídeo baseada em similaridade visual. Também serão apresentadas as funcionalidades oferecidas pelas ferramentas criadas para dar suporte ao ambiente proposto.

O estudo de caso aplicado ao ambiente foi direcionado para a categoria de vídeos jornalísticos e o objeto dos experimentos para tal categoria foi o **Jornal Nacional** da Rede Globo, sendo utilizados os vídeos jornalísticos que foram ao ar entre os meses de Setembro de 2007 e Fevereiro de 2008.

A Tabela 2 mostra as edições utilizadas, com as datas em que as mesmas foram ao ar e as durações aproximadas de cada uma delas. Todas as edições foram gravadas com resolução espacial de 320 x 240 *pixels* (1/4 de tela) e resolução temporal de 29.97 quadros por segundo.

Para obtenção dos vídeos foi utilizada uma placa de captura *Pinnacle* PCTV pro PCI, com uso de antena de TV interna comum. O uso desse tipo de antena apresentou um pequeno ruído branco nos vídeos, o que não foi considerado grave e até bem vindo no sentido em que reforça a eficiência dos resultados de busca por similaridade. Isso porque, mesmo com ruído, trechos de vinhetas iguais vão apresentar pequenas diferenças visuais ruidosas. Outra característica indesejável que foi notada nos vídeos está relacionada com uma alteração nos níveis de brilho e/ou de saturação de cor em edições capturadas em datas distintas. Como será mostrado nos resultados do capítulo 6, esse tipo de variação em brilho tem influencia direta na busca baseada em características estatísticas.

[5] No decorrer do texto, o termo ambiente engloba o conjunto das ferramentas de *parsing*, indexação, busca e navegação do vídeo.

Tabela 2 – Edições do Jornal Nacional usadas nos experimentos

| Data da Edição | Duração em Minutos | Data da Edição | Duração em Minutos | Data da Edição | Duração em Minutos |
|---|---|---|---|---|---|
| 26/9/2007 | 44,2 | 27/11/2007 | 45,4 | 26/1/2008 | 46,9 |
| 27/9/2007 | 35,4 | 28/11/2007 | 37,6 | 28/1/2008 | 43,9 |
| 28/9/2007 | 41,7 | 29/11/2007 | 36,5 | 29/1/2008 | 45,6 |
| 1/10/2007 | 38,4 | 30/11/2007 | 43,3 | 30/1/2008 | 40,3 |
| 31/10/2007 | 40,1 | 3/12/2007 | 39,3 | 1/2/2008 | 40,5 |
| 2/11/2007 | 39,6 | 4/12/2007 | 45,1 | 2/2/2008 | 42,7 |
| 5/11/2007 | 42,1 | 5/12/2007 | 44,3 | 4/2/2008 | 11,9 |
| 13/11/2007 | 45,5 | 6/12/2007 | 45,1 | 6/2/2008 | 39,2 |
| 14/11/2007 | 35,0 | 7/12/2007 | 14,3 | 7/2/2008 | 46,9 |
| 15/11/2007 | 34,0 | 18/1/2008 | 43,0 | 8/2/2008 | 45,6 |
| 16/11/2007 | 38,8 | 19/1/2008 | 42,2 | 9/2/2008 | 43,5 |
| 20/11/2007 | 43,0 | 21/1/2008 | 41,2 | 11/2/2008 | 44,6 |
| 21/11/2007 | 31,1 | 22/1/2008 | 41,5 | 12/2/2008 | 46,1 |
| 22/11/2007 | 46,0 | 23/1/2008 | 30,0 | 13/2/2008 | 36,7 |
| 23/11/2007 | 43,3 | 24/1/2008 | 47,2 | | |
| 26/11/2007 | 46,0 | 25/1/2008 | 45,9 | | |

Fonte: elaborada pelo autor (2008).

Os vídeos foram gravados do início ao fim com os comerciais locais da retransmissora TV Bahia. Mais especificamente, foram capturadas 46 edições do Jornal Nacional da Rede Globo, com duração média de 44 min por edição, incluindo os intervalos comerciais. Ao fim, um total de mais de 34 horas de vídeo e 3.420.835 quadros foram capturados.

## 5.1 O AMBIENTE DE INDEXAÇÃO E RECUPERAÇÃO DE CONTEÚDO

A arquitetura do ambiente desenvolvido foi dividida em dois módulos principais. O módulo que é responsável pelo *parsing* do fluxo do vídeo é chamado de **vídeo *parsing***. O outro módulo, que é responsável pela indexação, recuperação e navegação é denominado **vídeo oráculo**.

A Figura 28 ilustra, sucintamente, a interação das ferramentas entre si e com o usuário.

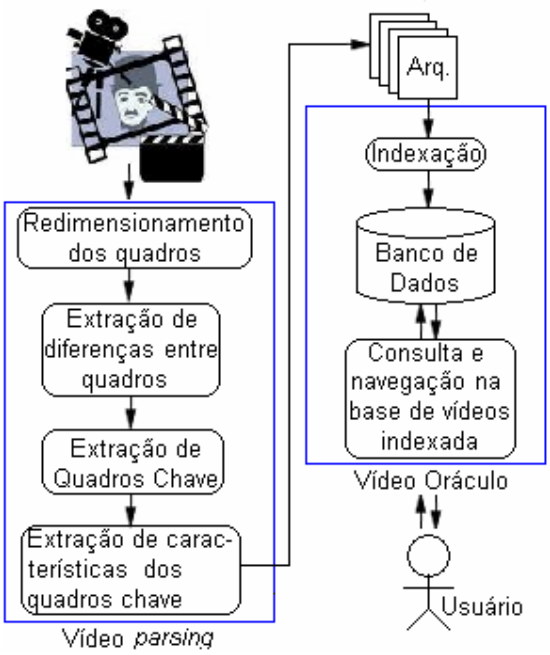


Figura 28 – Ambiente para indexação e recuperação de vídeo.

Fonte: elaborada pelo autor (2008).

### 5.1.1 Vídeo parsing

O vídeo *parsing* foi implementado em uma aplicação em C++ com recursos do Microsoft *DirectX©*. Qualquer vídeo cujo *codec* esteja instalado e reconhecido pelo *DirectX©* pode ter o *parsing* realizado. As funcionalidades do vídeo *parsing* que serão detalhadas nessa seção são:

a) redimensionamento dos quadros do vídeo para a resolução de 128 x 128 *pixels*;
b) conversão do espaço de cor dos quadros de RGB para YIQ;
c) extração de distância entre quadros, para dar suporte a implementação de detecção automática de corte de tomada, utilizando os métodos descritos em 2.7.1.1 e 2.7.1.2.: distância Euclidiana dos quadros em RGB; distância de histograma RGB e distância entre os quadros baseado na assinatura de *wavelet* descrita no ítem 4.8;
d) seleção de quadros-chave com armazenamento de uma cópia em arquivo JPG;
e) extração de características estatísticas dos quadros-chave;
f) extração da assinatura de *wavelet* dos quadros-chave (conforme item 4.8);

g) geração de arquivos com os dados estatísticos, as distâncias entre os quadros nas diferentes métricas e a assinatura de *wavelet* dos quadros-chave selecionados.

Nas subseções seguintes, cada uma das funcionalidades do vídeo *parsing* serão melhor detalhadas.

5.1.1.1 Redimensionamento de quadros

Todos os quadros do vídeo são processados pelo vídeo *parsing* com redução da resolução para 128x128 *pixels*. Três fatores motivaram a escolha dessa resolução mais baixa:

a) o processamento de quadros em menor resolução acarreta num custo computacional mais barato por envolver um conjunto de *pixels* mais reduzido;
b) a redução da resolução interfere pouco nos dados obtidos se comparados com a resolução original do quadro. As características estatísticas e de diferenças (ou distâncias) entre quadros não sofrem alteração relevante com a redução da resolução espacial;
c) a extração da assinatura de *wavelet* também sofre pouca interferência, já que os detalhes do quadro em maior resolução seriam naturalmente perdidos nesse processo. O motivo de se escolher o valor 128 é uma restrição do Algoritmo 2 da DWT mostrado no item 4.6.2.2. Essa restrição impõe que a quantidade de linhas e colunas sejam valores em potência de dois. Outros autores também fazem essa redução de resolução como no caso do trabalho de agrupamento de imagens dos autores Lui, Rosenberg e Rowley (2007).

O efeito da sub-amostragem é mostrado na Figura 29. Vale notar que o fato da razão de aspecto ser alterada tem como conseqüência uma distorção na imagem a ser posteriormente processada para extração de características visuais. Entretanto, isso acaba não sendo muito relevante no processo de busca de imagens por similaridade, uma vez que, as imagens utilizadas como parâmetro para as buscas sofrem o mesmo redimensionamento e, conseqüentemente, a mesma distorção.

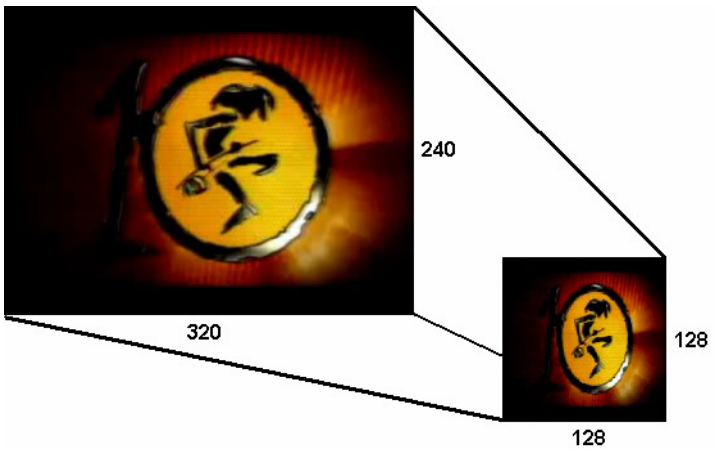


Figura 29 – Efeito da redução de resolução para 128x128 *pixels*

Fonte: elaborada pelo autor (2008).

Para o redimensionamento podem ser utilizados diversos algoritmos, tais como redimensionamento gaussiano, *bilinear*, *bspline*, *lanczos*, dentre outros. Contudo, na ferramenta de vídeo *parsing* foi usado o algoritmo de redução de resolução por sub-amostragem por ser uma técnica de baixo custo computacional e fácil implementação.

5.1.1.2 Conversão RGB para YIQ

Como será mostrado nas seções 5.1.1.5 e 5.1.1.7, as características visuais são extraídas dos quadros-chave no espaço de cor YIQ, assim, para dar suporte a esses procedimentos, os quadros que originalmente são obtidos pelo decodificador do vídeo (API do *DirectX*©) no espaço RGB precisam ser convertidos para o espaço YIQ. O processo de conversão é uma transformação linear bastante simples e largamente descrito na literatura, como por exemplo em Gonzalez e Woods (2000). Como na transformação do espaço de cor são gerados números reais no novo espaço YIQ, estes números cujas faixas de valores também são diferentes das encontradas no espaço RGB precisam de um reajuste com o objetivo de comportá-los na mesma estrutura de dados do espaço RGB. Outro motivo é a conformidade de escalas com mesmo peso para geração de um vetor que caracterize estatisticamente os quadros. Isso é descrito com mais detalhes no item 5.1.1.5.

Nas implementações do presente trabalho, cada um dos canais RGB variam numa faixa entre 0 e 255. No espaço YIQ, o canal “Y” é o único que varia nessa mesma faixa, sendo assim, os dados desse canal apenas são arredondados. Já o canal “I” varia entre -151.91 e 151.91. Desse modo, esses valores são re-escalados para uma faixa entre 0 e 255. O mesmo acontece com o canal “Q” que originalmente

varia entre -133.30 e 133.30. Com este processo, ao final, os três componentes do espaço de cor do quadro-chave no espaço YIQ são valores reais na faixa entre 0 e 255. Por fim, os dados dos canais “I” e “Q” também são arredondados.

5.1.1.3 Extração de distâncias entre quadros

Para a extração das distâncias entre quadros sucessivos, com objetivo de dar suporte à detecção automática de corte de tomada, foram testados três métodos:

a) a comparação de quadros *pixel*-a-*pixel* nos canais RGB conforme 2.7.1.1;
b) a comparação de quadros através dos histogramas dos canais RGB conforme 2.7.1.2;
c) a comparação das assinaturas dos quadros com base na eq. 12 usando os parâmetros de comparação de imagens da “busca por exemplo” conforme Tabela 1.

5.1.1.4 Seleção de quadros-chave

Devido à redundância temporal, quadros adjacentes tendem a apresentar características visuais muito similares. Conseqüentemente, as estatísticas e assinaturas de quadros por análise de *wavelet* são quase idênticas. Um dos objetivos deste trabalho é exatamente selecionar uma quantidade de quadros-chave que produzam a melhor relação entre baixa **redundância** *versus* baixa **perda** na captura de quadros, em outra palavras, deve produzir uma representação o mais reduzida possível do conteúdo do vídeo, mantendo as suas informações visuais não redundantes do conteúdo representado. Esses quadros-chave selecionados compõem um sumário que serve como base para extração de características visuais e, conseqüentemente, para a indexação de cada vídeo. Quadros consecutivos são considerados redundantes quando possuem grande similaridade visual quantificada por alguma medida de similaridade entre eles. Quadros perdidos são aqueles que deixam de ser capturados, criando uma lacuna num segmento temporal do vídeo sem ao menos um quadro que o represente. Ambas as variáveis (taxas de perda e redundância) são influenciadas pela escolha adequada do valor de um limiar utilizado pelo método de comparação de similaridade entre quadros. Se o limiar é baixo demais, muitos quadros redundantes são gerados; por outro lado, se é alto

demais, muitos podem ser perdidos. A Figura 30 ilustra o problema com seqüências de quadros-chave obtidas a partir de um vídeo jornalístico.

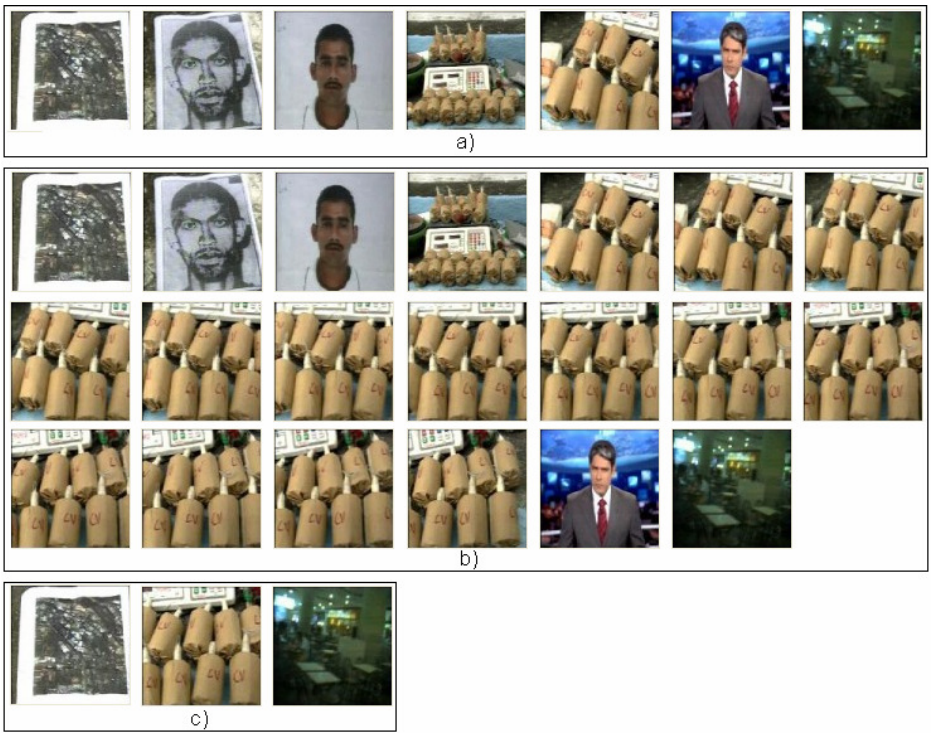


Figura 30 – Quadros-chave extraídos de um telejornal.

Fonte: elaborada pelo autor (2008).

Na Figura 30 (a) é apresentado um subconjunto da seleção de quadros-chave do modelo de referência, selecionados manualmente. A Figura 30 (b) mostra o resultado de uma seleção automática de quadros-chave com alto nível de redundância e baixa taxa de perdas, por fim a Figura 30 (c) ilustra o resultado de uma seleção automática com baixa redundância, mas com perda de quadros-chave.

A seleção de quadros-chave cumpre dois papéis fundamentais para a ferramenta de vídeo *parsing*: primeiro, a seleção de quais quadros serão processados com a extração de características e da assinatura de *wavelet*; segundo, a geração de um conjunto de imagens que serão utilizadas para sumarização e navegação do vídeo.

Vale destacar que a seleção de quadros-chave depende dos objetivos da aplicação que a utiliza. Para a aplicação de sumarização e suporte à busca de quadros orientada ao conteúdo visual, considera-se que a captura de quadros-chave deve apresentar, simultaneamente, a menor perda de quadros e a menor redundância possível, gerando um *static storyboard* do vídeo adequado ao suporte da implementação da busca por conteúdo visual com uma precisão aceitável. A

dificuldade aqui é que a redundância e a perda de quadros estão intimamente relacionadas, de modo que o ajuste de uma delas afeta a outra, assim, a questão passa a ser a escolha do limiar adequado.

A análise de um conjunto reduzido de quadros a um conjunto redundante traz basicamente três vantagens: maior velocidade de extração das características dos quadros pela ferramenta vídeo *parsing*; redução da quantidade de informação armazenada para indexação do vídeo e mais desempenho nas consultas realizadas através do vídeo oráculo.

Os resultados obtidos e os testes realizados para obtenção de quadros-chave são apresentados no capítulo 7 deste trabalho.

#### 5.1.1.5 Extração de Características Estatísticas

Uma vez selecionado um quadro-chave, um dos procedimentos da ferramenta de vídeo *parsing* é a extração das suas características estatísticas. São extraídas nove características invariantes, que serão mais bem detalhadas no decorrer dessa seção.

As características estatísticas são extraídas dos quadros convertidos para o espaço de cor YIQ. A escolha desse espaço de cor se dá por três motivos: Primeiro, a média dos valores dos *pixels* em cada canal YIQ dos quadros já é calculada como coeficiente global da transformada de *wavelet*, assim esse dado é reaproveitado reduzindo o custo de processamento e armazenamento. Segundo, o espaço YIQ separa os canais de luminância e crominância e terceiro, para manter um grau de conformidade no espaço de cor utilizado nas duas formas principais de busca, por vetor de características estatísticas e por assinatura de *wavelet*.

Um total de nove características foram selecionadas para composição de um vetor, que representa estatisticamente o conteúdo visual de cada quadro-chave. De fato, essas nove características se resumem conceitualmente a apenas três, já que cada uma foi aplicada nos três canais do espaço YIQ. Essas três características básicas extraídas são: média, entropia e variância (segundo momento do histograma), descritas nos itens 2.2.3 e 2.2.4. Embora as medidas citadas sejam indicadas por Gonzalez e Woods (2000) como sendo descritores de textura e aplicados como características locais, nesse trabalho elas foram usadas como descritores globais dos quadros. Outros trabalhos também fazem o uso de

características semelhantes, como é o caso do dos autores Faloutsos *et al.,* (1994) e Feng, Siu e Hong (2003).

Como a própria ferramenta de vídeo *parsing* sumariza automaticamente os vídeos, nem todos os quadros são processados para extração de características estatísticas. A medida que o vídeo *parsing* elege um novo quadro-chave, então esse quadro é analisado, suas características estatísticas são extraídas e os resultados são armazenados em arquivo de texto, sendo um vetor 9-dimensional por quadro-chave. Este vetor de características é representado pelos campos mostrados a seguir:

| med_y | med_i | med_q | ent_y | ent_i | ent_q | m2_y | m2_i | m2_q |
|---|---|---|---|---|---|---|---|---|

onde:

med_y - representa a média dos valores de *pixel* no canal Y;
med_i - representa a média dos valores de *pixel* no canal I;
med_q - representa a média dos valores de *pixel* no canal Q;
ent_y - representa a entropia dos *pixels* do canal Y;
ent_i - representa a entropia dos *pixels* do canal I;
ent_q - representa a entropia dos *pixels* do canal Q;
m2_y - representa o segundo momento do histograma do canal Y;
m2_i - representa o segundo momento do histograma do canal I;
m2_q - representa o segundo momento do histograma do canal Q.

As razões para escolha dessas três medidas foram as seguintes:

a) a cor pode ser usada com sucesso na discriminação de diversas entidades. A própria natureza emprega a cor de forma eficiente relacionado-a com uma classe de indivíduos ou identificando objetos de acordo com os seus pigmentos. Dessa forma uma série de mensagens são repassadas, como por exemplo, o perigo em animais venenosos que apresentam amarelo e preto alternados como padrão de cor (SWAIN; BALLARD, 1991);
b) a cor (e por conseguinte, a média aritmética dos canais de cor) é a primitiva de imagem largamente utilizada nos sistemas CBIR (FENG; SIU; ZHANG, 2003), como em (LI; WANG, 2003; DESELAERS, 2003; FLICKNER *et al*., 1997; IBM..., 2007; JACOBS; FINKELSTEIN;

SALESIN, 1995; DESELAERS; KEYSERS; NEY, 2004; MA; MANJUNATH, 1999);

c) o uso das três dimensões apresenta um potencial superior de discriminação ao uso de apenas um canal; por isso, foram extraídas três médias no presente trabalho, uma para cada canal de cor.

A cor, a textura e a forma são as principais características usadas em sistemas de CBIR (FENG; SIU; ZHANG, 2003). Com relação à textura, as demais características do vetor estatístico estão relacionadas com a discriminação dessa entidade.

5.1.1.6 Re-escalando os dados estatísticos

Como as faixas de variação em cada uma das três medidas estatísticas (média, entropia e variância) são diferentes, estas demandam pesos diferenciados na comparação entre vetores obtidos dos quadros. Portanto para deixar todas as medidas estatísticas com mesmo peso, as três foram alinhadas numa mesma faixa, desse modo, normalizadas, cada uma passa a ter o mesmo peso na comparação com a distância Euclidiana entre vetores que representam quadros-chave. No presente trabalho, originalmente a faixa de variação em cada medida era a seguinte: média [0-255]; entropia [0-8] e variância [0-128]. Todos os valores foram re-escalados para a faixa [0-255], ficando as medidas de entropia e variância em conformidade com a escala de médias dos valores de *pixel*. Os valores após re-escalados, também são arredondados, desse modo todo vetor é composto por medidas que podem ser codificadas em um byte cada. Conseqüentemente, um vetor estatístico representa cada quadro com apenas 9 bytes.

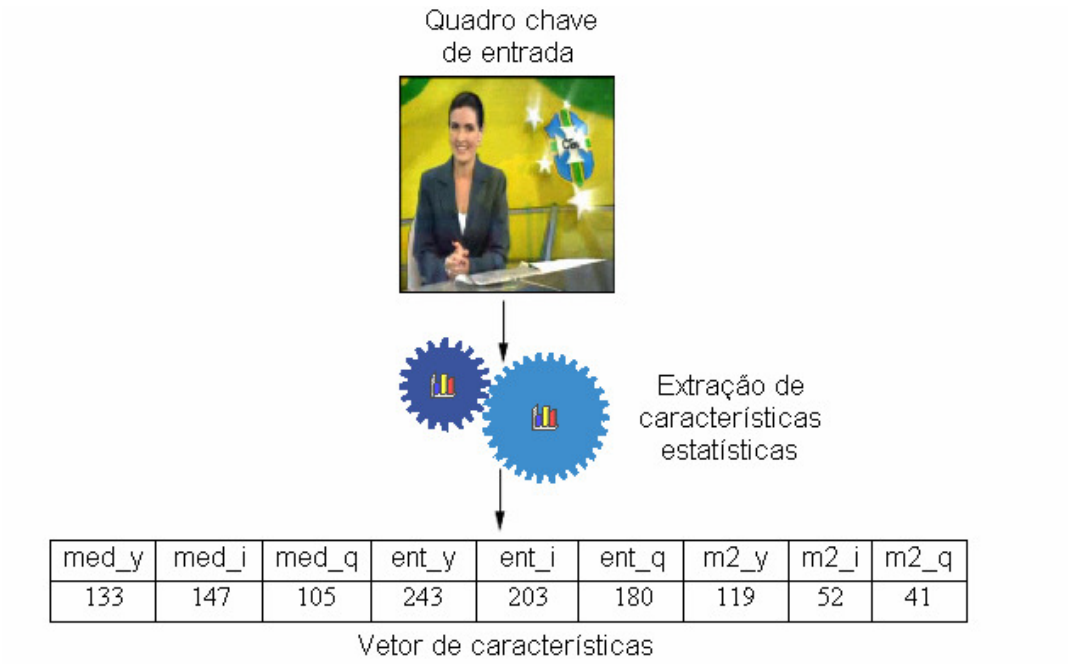


Figura 31 – Extração de características estatísticas dos quadros-chave.

Fonte: elaborada pelo autor (2008).

A Figura 31 ilustra sucintamente o processo de extração de características de um quadro-chave e sua representação vetorial em características estatísticas.

### 5.1.1.7 Extração da assinatura *Wavelet*

Outro procedimento realizado pela ferramenta de *parsing* sobre um quadro-chave é a extração da assinatura da imagem descrita no item 4.8.

Embora o uso de características de cor ajude na recuperação de conteúdo visual, apenas esse tipo de característica pode não ser interessante quando se procura conteúdos com maior valor semântico, ou seja, quando não apenas a ocorrência de determinadas cores ou texturas é importante, mas também a forma e a posição de alguns objetos. Isso fica bem representado na citação de Biederman (BIEDERMAN, 1986).

> *"...Surface characteristics such as color and texture will typically have only secondary roles in primal access...we may know that a chair has a particular color and texture simultaneously with its volumetric description, but it is only the volumetric description that provides efficient access to the representation of CHAIR..."*

Esse argumento de Biederman mostra que embora as características de cor continuem sendo importantes, essas características associadas à cor e/ou textura têm um papel secundário na discriminação de conteúdo, uma vez que um objeto do

mesmo tipo pode ter cores e texturas diversas. Seguindo esse raciocínio, a busca baseada em *wavelet* apresenta menor peso na informação de cor e mais peso na forma e posição dos objetos no quadro do vídeo. Desse modo, as características de cor, cujo papel é secundário na discriminação de conteúdo visual, na busca baseada em assinatura de *wavelet* tem um papel como filtro primário de quadros, em seguida os resultados provenientes desse filtro são refinados com os coeficientes desta assinatura, coeficientes cuja relação está principalmente na posição dos objetos na cena. A Figura 32 ilustra de modo simplificado a extração da assinatura do quadro. Esta assinatura é formada pela média dos valores dos *pixels* nos canais YIQ, pelas coordenadas dos 10 maiores coeficientes em cada canal e pelos 10 menores coeficientes também em cada canal do espaço de cor. No total são 60 posições de coeficientes. Os valores apresentados na Figura 32 são meramente ilustrativos, uma vez que seus pares de coordenadas estão codificados.

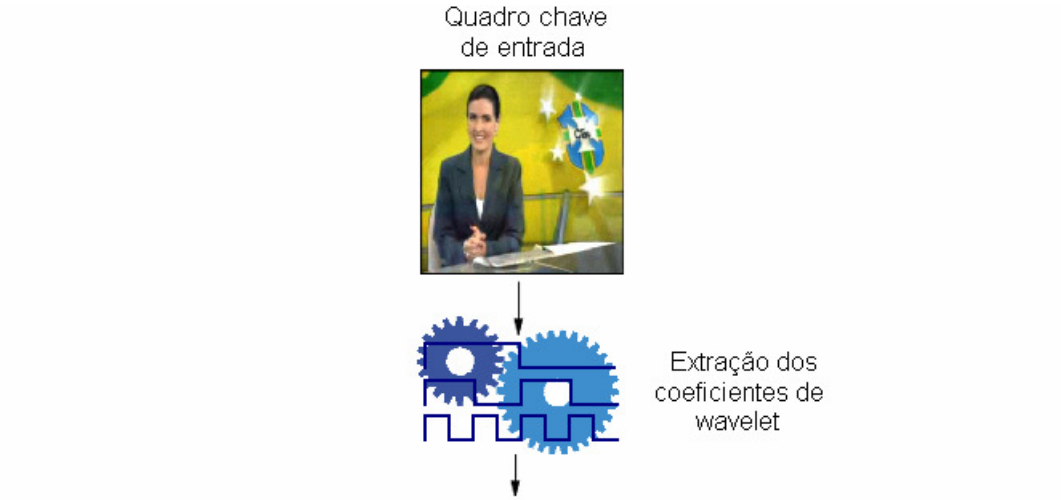


Figura 32 – Extração de assinatura de *wavelet* dos quadros-chave.

Fonte: elaborada pelo autor (2008).

Os detalhes do processo de extração da assinatura de *wavelet* foram descritos no item 4.8.

5.1.1.8 Geração de arquivos de dados

A ferramenta de vídeo *parsing* gera quatro arquivos com os dados resultantes do processamento. O primeiro contém as características do vídeo, tais como a taxa

de quadros por segundo, resolução, quantidade total de quadros, etc. O segundo arquivo contém a diferença inter-quadros para dar suporte à detecção de corte de tomada. O terceiro arquivo guarda as características estatísticas dos quadros-chave e o último arquivo armazena as assinaturas de *wavelet* dos quadros-chave. Esses dois últimos arquivos servem de base para a busca de quadros baseada em similaridade visual.

### 5.1.2 Vídeo Oráculo

A ferramenta vídeo oráculo, construída no ambiente de programação *Borland* C++ *builder* 6.0, "consome" os dados produzidos pelo vídeo *parsing* no formato de arquivos e "alimenta" um banco de dados com o resultado do seu processamento. Também é função do vídeo oráculo a análise destes dados, busca de quadros e navegação nos vídeos indexados. A seguir são apresentadas as principais funcionalidades do vídeo oráculo:

a) indexação e inserção dos dados gerados pela ferramenta de *parsing* (arquivos de texto) no banco de dados;
b) análise dos dados das distâncias oferecidas pelo vídeo *parsing* para detecção de cortes de tomada;
c) interface de uso para busca de quadros dos vídeos indexados;
d) busca por métrica estatística dos quadros de vídeo a partir de imagem fornecida como exemplo;
e) busca de quadros de vídeo por métrica de assinatura de imagem a partir de imagem fornecida da "busca por exemplo" ou " busca por rascunho";
f) acesso ao segmento do vídeo onde um quadro-chave tenha sido encontrado, independente da métrica de busca utilizada.

As seções seguintes deste capítulo detalham cada uma das funcionalidades anteriores do vídeo oráculo.

#### 5.1.2.1Indexação e inserção dos dados no banco de dados

O processo de inserção dos dados gerados pelo vídeo *parsing* é semi-automático, uma vez que o usuário seleciona individualmente qual vídeo previamente processado no *parsing* será indexado no repositório. Após selecionar

os dados do vídeo, cada um dos quatro arquivos será automaticamente inserido em sua respectiva tabela no banco de dados.

5.1.2.2 Detecção de cortes de tomadas

No caso dos dados de distância inter-quadros, o vídeo oráculo já analisa e identifica os cortes de tomada inserindo no banco de dados apenas as fronteiras das tomadas conforme distancias inter-quadros *pixel*-a-*pixel*. A justificativa do uso dessa métrica é mostrada no item 6.2 com os resultados de desempenho para detecção automática de cortes de tomada.

5.1.2.3 Interface para busca de quadros do vídeo

Para sistemas de recuperação de imagem, a interação entre o usuário e o sistema é crucial, uma vez que a informação de entrada é flexível e modificações na consulta só podem ser obtidas pelo envolvimento do usuário com o procedimento de pesquisa. Normalmente, os ambientes de sistemas de recuperação têm uma área para especificar a pesquisa e outra para exibir os resultados. No vídeo oráculo essa interface é bem simples: a especificação da pesquisa é feita selecionando-se um arquivo de imagem no diretório de arquivos. A interface do vídeo oráculo é mostrada na Figura 33. Especificamente na Figura 33 – (1) é mostrada uma imagem de busca e os resultados aparecem na Figura 33 – (5).

A especificação da pesquisa diz respeito aos parâmetros que serão utilizados na busca. No caso do presente trabalho, que tipo de imagem está sendo utilizada, se é um “rabisco” ou um “exemplo”, e qual a metodologia de busca será realizada, se por características estatísticas ou se por assinatura de *wavelet*. Esse último ainda subdividido em duas categorias: “busca por exemplo” ou “ busca por rascunho”.

A imagem é automaticamente identificada como um rascunho ou um exemplo, pela sua medida de entropia descrita na eq. 6 do item 3.2.4. Apesar da base logarítmica para o cálculo da entropia ser arbitrária, foi escolhida a base 2, pelo fato do resultado ser a quantidade média de bits para representação de cada *pixel* da imagem. O limiar usado para separar imagens de rascunho das imagens de exemplo foi o limiar de entropia igual a 3.3. Esse limiar foi obtido experimentalmente testando-se várias imagens de rascunho e de exemplo e a taxa de acertos foi de 100%.


Figura 33 - Interface de busca da de quadros.

Fonte: elaborada pelo autor (2008).

## 5.2 PROCESSO DE BUSCA DE QUADROS

Para efetuar uma busca, independente do método (estatístico ou por assinatura de *wavelet*), uma seqüência de passos mostrada na Figura 34. Primeiro, o usuário fornece uma imagem de “busca por exemplo” ou “busca por rascunho” ao sistema, essa imagem é chamada de “Imagem de Busca” *Q*. De acordo com o método de busca selecionado, a imagem terá suas características extraídas. O mesmo processo aplicado ao quadro no vídeo *parsing* é aplicado à imagem de busca, como o redimensionamento para 128x128 *pixels* e a conversão do espaço de cor de RGB para YIQ. Se for selecionado o método estatístico, serão extraídas as características estatísticas, compondo um vetor de nove dimensões, conforme descrito no item 5.1.1.5. Se o escolhido for o método de assinatura de *imagem*, a mesma será extraída de acordo com o item 5.1.1.7.

Uma vez extraídas as características da imagem de busca, já seria possível comparar o vetor ou a assinatura com toda a base de dados de quadros-chave e classificá-los. Entretanto, tal operação, em toda base, envolve um alto custo computacional. Para contornar esse problema, um filtro pré-seleciona os vetores mais prováveis a representarem as imagens alvo que se deseja, trazendo um subconjunto de vetores muito menor para ser classificado, um processo semelhante ao realizado no QBIC (FLICKNER *et al*., 1997). Desse modo, o banco de dados recebe uma instrução para selecionar apenas os dados dos quadros com características dentro de uma determinada faixa de valores. Essa faixa é definida tanto com base nas características da imagem de busca quanto no método de busca utilizado. O objetivo desse procedimento é justamente descartar os quadros cujas características não tenham absolutamente nada a ver com a imagem de busca, por exemplo, se a imagem de busca tem média de intensidade de luminância Y igual a 10 unidades, não faz sentido classificar os vetores cuja média de luminância é igual a 250, pois estes em princípio, já são completamente distintos da imagem de busca.

Para o filtro primário, na busca baseada em características estatísticas, são estabelecidos dois vetores, um vetor piso outro teto. Os dois vetores são construídos com base nos valores obtidos pelo vetor extraído da imagem de busca, ou vetor base. Para gerar o vetor piso, subtraem-se 35 unidades em cada característica estatística do vetor base. Já para o vetor teto, cada característica recebe +35 unidades. Desse modo, o banco de dados retorna todos os vetores que obedeçam à faixa das características estatísticas entre o vetor base e teto. O exemplo a seguir ilustra a composição de dois vetores característicos:

Vetor Teto

| med_y | med_i | med_q | ent_y | ent_i | ent_q | m2_y | m2_i | m2_q |
|---|---|---|---|---|---|---|---|---|
| 168 | 182 | 140 | 278 | 238 | 215 | 154 | 87 | 76 |

**Vetor Base**

| med_y | med_i | med_q | ent_y | ent_i | ent_q | m2_y | m2_i | m2_q |
|---|---|---|---|---|---|---|---|---|
| 133 | 147 | 105 | 243 | 203 | 180 | 119 | 52 | 41 |

Vetor Piso

| med_y | med_i | med_q | ent_y | ent_i | ent_q | m2_y | m2_i | m2_q |
|---|---|---|---|---|---|---|---|---|
| 98 | 112 | 70 | 208 | 168 | 145 | 84 | 17 | 6 |

A definição da faixa de -35 a +35 unidades em cada característica foi obtida de forma heurística e experimental a fim de que esse filtro traga, além dos quadros

de fato visualmente similares, um conjunto limitado de quadros com um certo grau de dissimilaridade visual, para que nenhum quadro possivelmente similar seja descartado na classificação.

Para os métodos de busca baseados em assinatura de imagem, o filtro é aplicado apenas nas características de média dos valores de *pixels*, com as seguintes definições: para a "busca por exemplo", o canal de luminância Y recebe -25 e +25 unidades para o piso e teto respectivamente e -20 e +20 para os pisos e tetos nos canais de crominância IQ. Já no filtro primário da busca "por rascunho" os valores são de -60 e +60 para o canal Y e -50 e +50 para os canais IQ. O motivo de empregar uma faixa muito maior no filtro primário nas componentes de média da métrica baseada na "busca por rascunho" é que o usuário pode criar rabiscos com cores e intensidade de brilho com maior grau de liberdade. Assim, o filtro traz um conjunto de assinaturas de quadros bem maior, que será refinado em termos de comparação de similaridade visual de acordo com os coeficientes da transformada de *wavelet* na equação de comparação de assinatura descrita no item 4.8.2.

Após a filtragem, cada vetor/assinatura representativo de um quadro resultante é comparado com as características da imagem de busca, de acordo com sua métrica apropriada. Se for o método estatístico, é utilizada a métrica de distância Euclidiana, conforme item 2.7.1.1. Se for o método de *wavelet*, é utilizada uma das métricas de comparação de assinatura mostrada no item 0 para "busca por rascunho" ou "busca por exemplo".

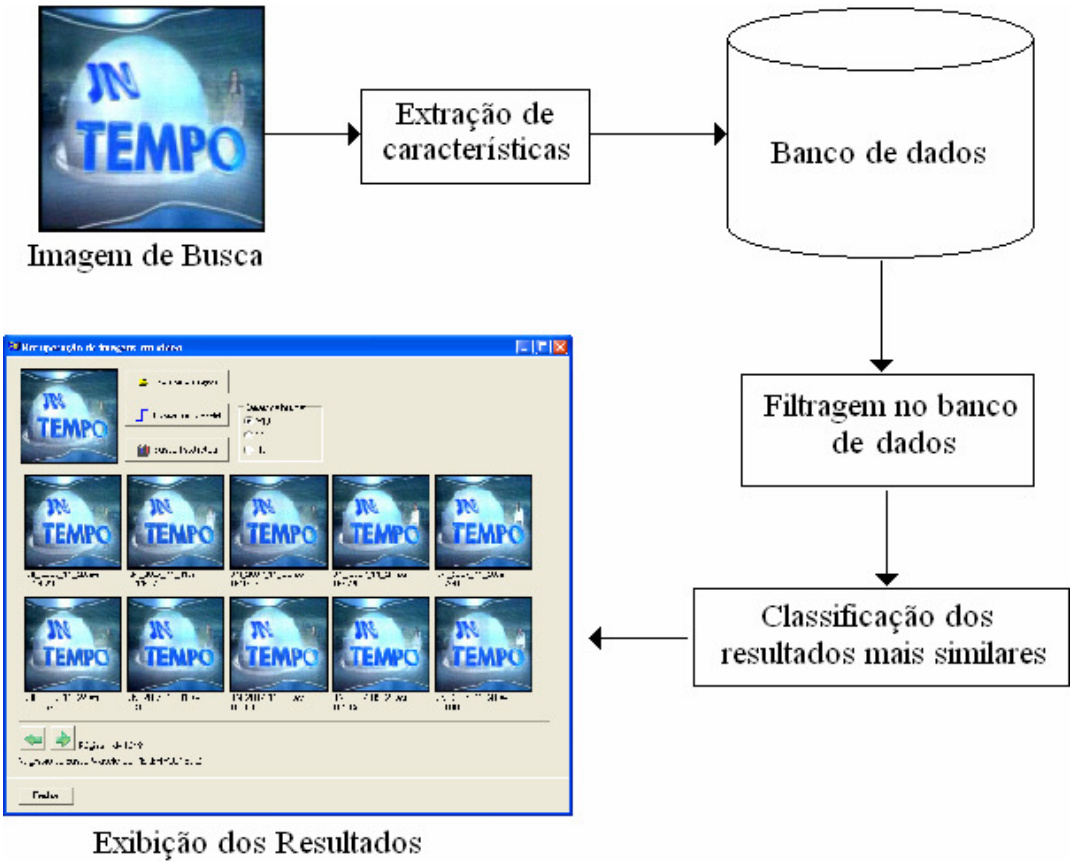


Figura 34 – Processo de busca de quadros.

Fonte: elaborada pelo autor (2008).

Finalmente, uma vez comparadas as distâncias, os vetores são classificados de forma ascendente através do algoritmo de ordenação *QuickSort* (ZIVIANI *et al.*, 2004) e os resultados, após classificados por ordem de similaridade, são exibidos ao usuário.

### 5.21 Acesso ao ponto exato do vídeo correspondente ao quadro de busca

No vídeo oráculo há um *thumbnail* para cada quadro-chave classificado no processo de busca. Os *thumbnails* são mostrados ordenados em páginas, sendo 10 quadros por página. A Figura 33 – (6) ilustra um exemplo de resultado do processo realizado pelo vídeo oráculo na busca e exibição dos quadros-chave mais similares. Abaixo de cada *thumbnail* é mostrada uma legenda informado o título do vídeo, e a posição temporal de ocorrência deste quadro-chave. Uma vez encontrado o quadro alvo que pertença a uma cena que o usuário deseja visualizar, basta clicar sobre o *thumbnail* que o vídeo oráculo automaticamente acessa e exibe o segmento temporal relativo ao quadro-chave.

## 5.3 CONSIDERAÇÕES SOBRE OS OBJETIVOS PROPOSTOS PARA OS DIVERSOS TIPOS DE BUSCAS DE QUADROS

O vídeo oráculo tem a pretensão de atender aos diversos objetivos de busca visual de quadros em vídeos. Desse modo, se faz primeiramente necessário definir os objetivos de busca do ponto de vista conceitual, independente da tecnologia para alcançá-lo. Quanto a esses objetivos foram considerados os seguintes:

**Busca de quadros mais similares** – Considera-se nesse trabalho, que essa busca tem o objetivo de recuperar os quadros mais similares ao quadro fornecido, admitindo-se um grau de tolerância menor do que seria admitido na **busca dos quadros com tolerância a mudanças**. Enquadram-se no objetivo desse tipo de busca quadros que *a priori* se sabe pertencerem a uma categoria que pode aparecer diversas vezes na base de quadros-chave, tais como quadros presentes em vinhetas e comerciais que se repetem durante a programação. Ao fazer esse tipo de busca, o usuário deve dispor de um quadro de exemplo que seja muito semelhante ao que se pretende recuperar. Em outras palavras, o usuário deseja recuperar todas as re-ocorrências das cenas cujo quadro-chave ele dispõem como exemplo. Uma ilustração desse tipo de busca seria a seleção de todos os segmentos de previsão do tempo, baseado em um quadro comum da mini-vinheta que chama este tipo de noticiário no telejornal, como pode ser visto na Figura 34.

**Busca de quadros similares com tolerância a mudanças** – Nesse tipo de busca há um maior grau de liberdade na tolerância dada à similaridade. Esses casos podem ser úteis em diversos casos, como exemplo típico, observado nos vídeos do Jornal Nacional, o apresentador âncora costuma ter um cenário de fundo temático com a notícia. Por exemplo, notícias relacionadas à política têm um fundo temático próprio, assim como as notícias relacionadas à saúde, educação, economia, esportes, futebol, notícias internacionais e assim sucessivamente. Alguns exemplos de imagens similares com tolerância a mudanças podem ser observadas na Figura 35.

Os experimentos mostraram que nesse tipo de busca, uma tolerância na margem de similaridade deve ser considerada para atender casos como os encontrados nos vídeos estudados. Nestes vídeos, embora o plano de fundo temático à notícia, siga um padrão relativamente bem definido, de acordo com a notícia, tanto o próprio apresentador pode mudar em edições diferentes do tele-

jornal, quanto a cor do terno ou outros objetos. Nesse tipo de busca, a similaridade visual tenta ser relacionada de alguma forma com o conteúdo semântico dos quadros.


Figura 35 – Exemplos de quadros similares com tolerância a mudanças.
Fonte: elaborada pelo autor (2008).

Um outro exemplo acontece nas notícias internacionais, onde além das mudanças anteriormente descritas, a bandeira do país em questão é alterada no fundo. A Figura 35 ilustra esse exemplo de busca de quadros similares com tolerância a mudanças, na Figura 35a embora todos os quadros sejam chamadas de notícias internacionais, há mudanças no apresentador, na cor do terno, no mapa e na bandeira do país ao qual a notícia se refere. Vale notar, no caso da bandeira, que, tanto seu tamanho quanto as suas cores e formas variam bastante de um país para outro, e mesmo assim, esse tipo de busca foi bem sucedida nos experimentos realizados. Outras buscas envolvendo pequenas mudanças podem ser traçadas pelo usuário. Assim, faz-se necessário esse grau de tolerância a mudanças em cor e formato de alguns objetos para que os quadros alvo sejam alcançados com sucesso.

Neste trabalho são considerados dois tipos de similaridade: a **similaridade puramente visual** (baseada nas características extraídas dos quadros, sem considerar a semântica do conteúdo) e a **similaridade visual semântica**. Se apenas a similaridade puramente visual é levada em conta, quadros podem ser considerados similares quando possuírem similaridade em termos de cor, nível de

entropia e contraste, mesmo quando possuírem objetos completamente descorrelacionados. No caso da similaridade visual semântica o conteúdo semântico dos quadros deve ser muito similar, ou seja, os mesmos objetos e/ou o mesmo contexto devem ser encontrados nos quadros analisados. Por exemplo, se numa busca deseja-se encontrar o apresentador âncora do telejornal, os quadros com similaridade visual semântica poderiam ser aqueles que contêm qualquer apresentador âncora de telejornal. Por outro lado, os quadros com similaridade puramente visual poderiam ser aqueles nos quais aparecem objetos com cores e contraste semelhante as do fundo do telejornal (e, portanto, com conteúdos descorrelacionados do ponto de vista semântico).

É importante deixar claro dois pontos. Primeiro, a similaridade semântica é extremamente subjetiva e seu conceito pode variar bastante de um indivíduo para outro. E segundo, o ambiente proposto no presente trabalho não implementa nenhuma técnica de inteligência artificial para identificação de conteúdo semântico (as buscas foram implementadas tendo como base as técnicas descritas anteriormente). Os resultados desses tipos de buscas são apresentados no capítulo 6.

# 6 AVALIAÇÃO DO AMBIENTE DE RECUPERAÇÃO DE VÍDEOS PROPOSTO

Este capítulo apresenta os resultados dos diversos experimentos realizados com as ferramentas vídeo *parsing* e vídeo oráculo. Os resultados dos experimentos tiveram impacto em várias configurações implementadas na versão final das ferramentas de vídeo *parsing* e vídeo oráculo, descritas no capítulo anterior. Os experimentos realizados foram os seguintes: (i) comparação nas relações de perda *versus* redundância de quadros-chave para sumarização dos vídeos jornalísticos; (ii) comparação entre métodos para detecção de corte de tomada e (iii) comparação de desempenho nas buscas de quadros-chave.

## 6.1 AVALIAÇÃO DE MÉTODOS DE SUMARIZAÇÃO DE VÍDEOS

A primeira avaliação de desempenho trata dos resultados experimentais obtidos no processo de extração de quadros-chave de vídeos jornalísticos para sumarização e indexação propostos no item 5.1.1.4. Os métodos baseados em diferença entre quadros para os espaços RGB e YIQ permitiram a criação de quatro abordagens básicas para extração de quadros-chave de um vídeo:

a) comparação *pixel* a *pixel* no espaço RGB;
b) comparação de histograma no espaço RGB;
c) comparação *pixel* a *pixel* nos canais I e Q do espaço de cor YIQ;
d) comparação de histograma nos canais I e Q do espaço de cor YIQ.

Além da aplicação isolada de cada uma das abordagens básicas, o presente trabalho propõe e analisa a possibilidade de melhorar a relação perda *versus* redundância no processo de captura de quadros-chave através da integração dessas abordagens. As três novas abordagens correspondem à combinação dos resultados obtidos pelos métodos baseados nas diferenças *pixel*-a-*pixel* e de histogramas de quadros nos canais RGB (combinação de 1 e 2), nos canais I e Q (combinação de 3 e 4) e, finalmente, à combinação de todas as abordagens básicas (de 1 a 4).

### 6.1.1 Metodologia e avaliação de captura de quadros-chave

O principal critério de avaliação dos métodos considerado nesta dissertação é a relação entre quadros redundantes gerados *versus* quadros perdidos durante o

processo de extração de quadros-chave. A avaliação foi baseada na comparação entre os resultados obtidos por cada uma das abordagens (em termos de quadros-chave produzidos e/ou perdidos) e aqueles gerados através da intervenção humana (e, portanto, tidos como ideais).

Os autores Drew e Au (2000) tratam a extração de quadros-chave fazendo uma comparação entre uma sumarização com quadros-chave construída manualmente e bases construídas automaticamente. Eles também avaliam taxas de perda e redundância. Contudo, além da base utilizada por eles ser bem menor (aproximadamente 10 mil quadros), os métodos empregados e os detalhes na avaliação da perda *versus* redundância são diferentes dos considerados no presente trabalho.

A seqüência de quadros-chave de referência do presente trabalho foi obtida por inspeção visual da edição do Jornal Nacional de 26/09/08, inclusos os intervalos comerciais. A amostra dessa edição tem duração de 44 minutos e um total de 79.432 quadros. Manualmente foram identificados 582 cortes de tomada e 883 quadros-chave, resultando numa média de 1,51 quadro-chave por tomada.

As métricas consideradas para a avaliação das abordagens são os percentuais médios de quadros redundantes (extras) gerados e dos quadros perdidos com relação aos quadros das tomadas da seqüência de referência manual. Por exemplo, se uma tomada contiver 1 quadro-chave na referência e 2 na abordagem automática, considera-se que a redundância é igual a 100% naquela tomada. Por outro lado, se uma tomada com 4 quadros na seqüência de referência estiver representada por 3 quadros na seleção automática, a medida de perda será de 25%. Os parâmetros de referência considerados são apresentados na Tabela 3.

Tabela 3 – Modelo de refêrencia de comparação de quadros-chave do vídeo 26/09/2007

| Limiar | KF/S | Perda | Red. | TotP | TotR | Total |
|---|---|---|---|---|---|---|
| Não definido | 1,51 | 0% | 0% | 0 | 0 | 883 |

Fonte: elaborada pelo autor (2008).

Como os resultados apresentados são aqueles esperados de um processo “ideal” de extração automática de quadros, não há perda nem redundância. Nessa tabela (e em todas as outras utilizadas na seqüência do presente trabalho) serão consideradas as definições mostradas a seguir:

Limiar - O limiar usado no método;

KF/S - A média de quadros-chave por tomada;

Perda - O percentual médio de perda de captura de quadros;

Red. - O percentual médio de redundância de captura de quadros;

TotP - A quantidade absoluta de quadros perdidos;

TotR - A quantidade absoluta de quadros redundantes;

Total - A quantidade total de quadros-chave capturados.

Embora os limiares definidos nos testes com o vídeo de referência possam variar de um vídeo para outro, os mesmos valores foram utilizados para extração dos quadros-chave das demais edições da base, por se tratar de uma mesma categoria de vídeo. Quanto às outras categorias, embora os limiares não sejam considerados perfeitamente válidos, considera-se importante a avaliação das características de cada método e a comparação entre os mesmos. Os resultados dessas comparações são apresentados e discutidos a seguir.

### 6.1.2 Resultados dos experimentos

Os resultados obtidos pelas quatro abordagens baseadas nas distâncias entre quadros para seleção de quadros-chave, bem como as três combinações propostas entre as mesmas são apresentados nas seções a seguir. O critério para a seleção do melhor limiar para cada abordagem é o seguinte: o melhor limiar é aquele para o qual as curvas de perda e redundância se cruzam no gráfico de desempenho da abordagem. Intuitivamente, fora desse ponto de intersecção, ou existe excesso de quadros-chave (muita redundância) ou falta de quadros-chave (muita perda). O cruzamento das curvas determina, então, o melhor limiar para a aplicação da extração automática de quadros-chave para o método considerado.

#### 6.1.2.1 Comparação de quadros *pixel*-a-*pixel* RGB

O método de comparação de quadros *pixel*-a-*pixel* no espaço de cor RGB, embora bastante simples e conhecido, foi o melhor dentre todos testados individualmente. A Tabela 4 mostra os resultados obtidos variando-se o limiar de captura entre 45 e 75. No estudo desse trabalho o limiar pode variar entre 0 e 255. Supondo dois quadros completamente diferentes, no pior caso, um quadro completamente branco outro completamente preto, a diferença média para a presente abordagem seria igual a 255, obviamente quadros idênticos têm diferença igual a 0.

Tabela 4 – Desempenho do método *pixel*-a-*pixel* RGB

| Limiar | KF/S | Perda | Red. | TotP | TotR | Total |
|---|---|---|---|---|---|---|
| 45 | 2,74 | 8,2% | 65,2% | 90 | 763 | 1596 |
| 52,5 | 1,94 | 16,4% | 33,2% | 158 | 380 | 1132 |
| 60 | 1,39 | 26,3% | 14,9% | 265 | 173 | 809 |
| 67,5 | 1,02 | 39,8% | 7,9% | 387 | 87 | 595 |
| 75 | 0,71 | 51,7% | 3,1% | 505 | 30 | 413 |

Fonte: elaborada pelo autor (2008).

A melhor relação perda *versus* redundância ocorre próximo a 22%, conforme o Gráfico 1, ou seja, há simultaneamente 22% de perda e 22% de redundância. O método tem como vantagem a robustez aos efeitos de corte de tomada com *fading* ou dissolução, evitando a geração de quadros redundantes. Porém, pequenos movimentos de câmera e/ou objetos podem provocar a geração excessiva de quadros redundantes, sendo este o seu ponto fraco.

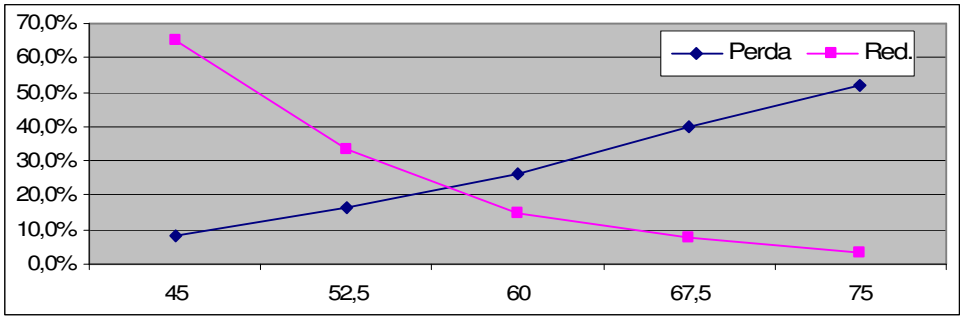


Gráfico 1 - Desempenho do método *pixel*-a-*pixel* RGB

Fonte: elaborado pelo autor (2008).

6.1.2.2Comparação de histogramas RGB

O método de comparação por histograma RGB apresenta características inversas à comparação *pixel*-a-*pixel*. Sua vantagem é a boa resistência aos efeitos de dissolução e sua principal desvantagem é a grande sensibilidade a efeitos de *fading*, capturando quadros em excesso nessas transições. Cenas com *flashes* fotográficos expõem outra fraqueza do método, uma vez que eles alteram bruscamente a luminosidade do quadro e, por conseguinte, o seu histograma. Uma característica importante do método é a sua maior resistência à movimentação de objetos e de câmera. Embora seja desejável uma resistência a pequenas movimentações, o método pode deixar de capturar quadros importantes cuja variação na posição dos objetos é relevante. A Tabela 5 mostra os resultados do

método variando-se o limiar de captura entre 0,10 e 0,175 em quatro experimentos. Sendo este limiar a distância entre dois histogramas conforme eq 3 em 2.7.1.2.

Tabela 5 - Desempenho do método de comparação de histograma RGB

| Limiar | KF/S | Perda | Red. | TotP | TotR | Total |
|---|---|---|---|---|---|---|
| 0,10 | 2,24 | 16,8% | 49,1% | 180 | 560 | 1294 |
| 0,125 | 1,67 | 27,8% | 30,4% | 277 | 345 | 966 |
| 0,15 | 1,29 | 41,3% | 20,6% | 394 | 241 | 746 |
| 0,175 | 1,07 | 48,2% | 16,6% | 460 | 191 | 621 |

Fonte: elaborada pelo autor (2008).

O Gráfico 2 mostra que a intersecção entre a perda e redundância acontece próximo a 30%.

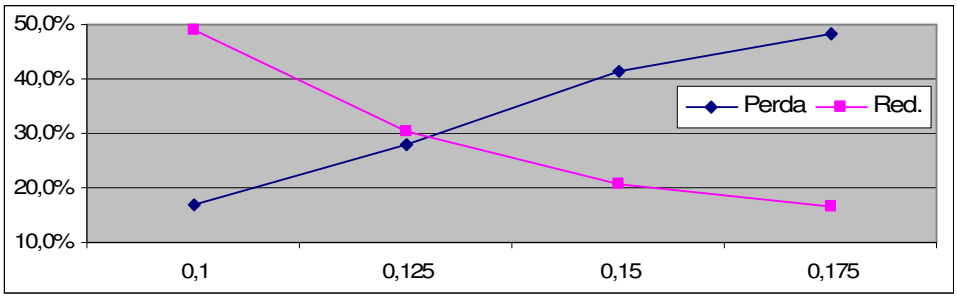


Gráfico 2 - Desempenho do método de comparação de histograma RGB

Fonte: elaborada pelo autor (2008).

6.1.2.3Comparação de quadros *pixel*-a-*pixel* nos canais IQ

Com características semelhantes ao método *pixel-a-pixel* em RGB e apresentando grande sensibilidade a movimentos de objetos e câmera, esse método ainda tem problemas ao tratar efeitos de dissolução, agregando um pouco mais de redundância aos resultados que no espaço RGB. A Tabela 6 mostra os resultados obtidos com a variação do limiar entre 12,5 e 22,5.

Tabela 6 - Desempenho do método *pixel*-a-*pixel* IQ

| Limiar | KF/S | Perda | Red. | TotP | TotR | Total |
|---|---|---|---|---|---|---|
| 12,5 | 3,34 | 20,0% | 100,0% | 183 | 1204 | 1951 |
| 15 | 2,24 | 32,4% | 52,4% | 290 | 677 | 1310 |
| 17,5 | 1,6 | 41,9% | 33,3% | 391 | 429 | 938 |
| 20 | 1,18 | 52,1% | 21,5% | 484 | 280 | 688 |
| 22,5 | 0,86 | 59,6% | 14,1% | 564 | 181 | 506 |

Como pode ser observado no Gráfico 3, o ponto de cruzamento da relação de perda por redundância acontece próximo a 40%, um valor relativamente alto, correspondendo ao pior desempenho entre os métodos estudados isoladamente.

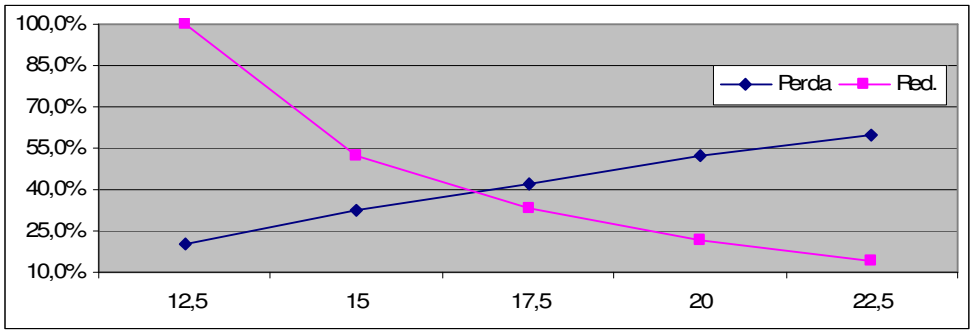


Gráfico 3 - Desempenho do método *pixel*-a-*pixel* IQ

Fonte: elaborado pelo autor (2008).

6.1.2.4Comparação de histogramas IQ

Semelhante ao método de comparação de histogramas em RGB, esse método é um pouco mais resistente a transições com *fading*. Em contrapartida, ele é bastante sensível aos efeitos de dissolução, capturando quadros em excesso, nessas transições de tomada. A resistência a movimentos de objetos e câmera também causa uma grande perda de quadros-chave do mesmo modo que no método baseado em histogramas RGB. A Tabela 7 apresenta os resultados do método com variação do limiar entre 0,25 e 0,40.

Tabela 7 - Desempenho do método de comparação de histograma IQ

| Limiar | KF/S | Perda | Red. | TotP | TotR | Total |
|---|---|---|---|---|---|---|
| 0,25 | 2,03 | 14,1% | 43,2% | 150 | 419 | 1175 |
| 0,30 | 1,55 | 24,2% | 26,0% | 244 | 244 | 899 |
| 0,35 | 1,2 | 36,1% | 18,9% | 358 | 156 | 696 |
| 0,40 | 0,92 | 45,6% | 9,5% | 437 | 77 | 534 |

Fonte: elaborada pelo autor (2008).

O Gráfico 4 mostra que o cruzamento das curvas de perda e redundância para a comparação de histogramas nos canais IQ acontece próximo a 25%, com um desempenho ligeiramente melhor que a comparação de histogramas em RGB.

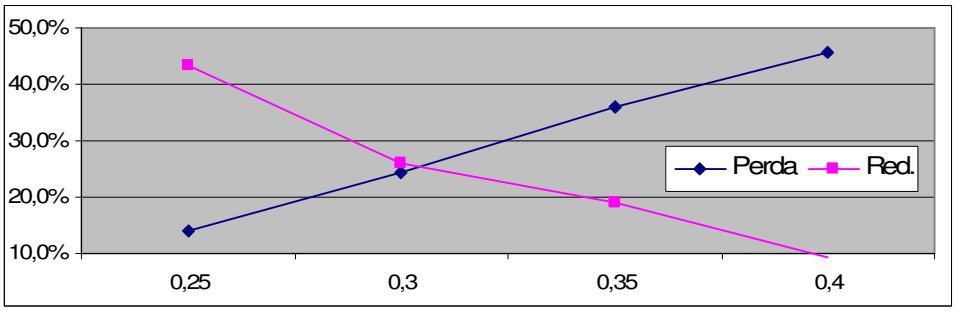


Gráfico 4 - Desempenho do método de comparação de histograma IQ

Fonte: elaborado pelo autor (2008).

### 6.1.3 Combinação de métodos

A combinação de métodos obriga apenas a uma pequena alteração no Algoritmo 1. Ao invés de apenas uma medida de distância isolada, são testadas as medidas de mais de um método, com a operação relacional “E” ou “AND” na condição “SE” da linha 4 para cada método adicionado.

#### 6.1.3.1 Combinação dos métodos de histograma e diferença *pixel*-a-*pixel* RGB

A combinação dos métodos de histograma e comparação *pixel*-a-*pixel* em RGB apresentou um dos melhores resultados dentre todos os métodos avaliados. Não há vulnerabilidade em efeito de transição de tomada com *fading* ou dissoluções; também há uma boa captura de quadros-chave em cenas em branco e preto, o que não acontece nos métodos de histograma e *pixel*-a-*pixel* IQ. A vulnerabilidade da combinação desses métodos aparece em cenas com *flash*es, onde a variação de luz causa mudanças bruscas no histograma e em nível de *pixel*.

A Tabela 8 mostra os diversos testes realizados. Na coluna inicial, o primeiro valor indica o limiar da diferença *pixel*-a-*pixel* e o segundo, o da distância de histogramas RGB.

Tabela 8 – Métodos de histograma e diferença *pixel*-a-*pixel*

| Limiares | KF/S | Perda | Red. | TotP | TotR | Total |
|---|---|---|---|---|---|---|
| 30-0,07 | 2,48 | 5,8% | 53,5% | 68 | 598 | 1449 |
| 40-0,08 | 1,85 | 12,2% | 26,6% | 138 | 310 | 1079 |
| 45-0,07 | 1,74 | 13,6% | 21,8% | 149 | 259 | 1014 |
| 45-0,08 | 1,65 | 15,5% | 19,3% | 171 | 230 | 963 |
| 45-0,09 | 1,52 | 17,2% | 16,0% | 200 | 187 | 888 |
| 50-0,07 | 1,55 | 18,0% | 16,0% | 179 | 187 | 907 |
| 50-0,08 | 1,48 | 19,3% | 14,3% | 201 | 164 | 860 |
| 50-0,09 | 1,38 | 22,0% | 12,1% | 229 | 134 | 803 |
| 55-0,07 | 1,36 | 23,2% | 11,8% | 236 | 132 | 793 |

Fonte: elaborada pelo autor (2008).

O Gráfico 5 mostra que neste método o cruzamento da perda com a redundância acontece em aproximadamente 17%.

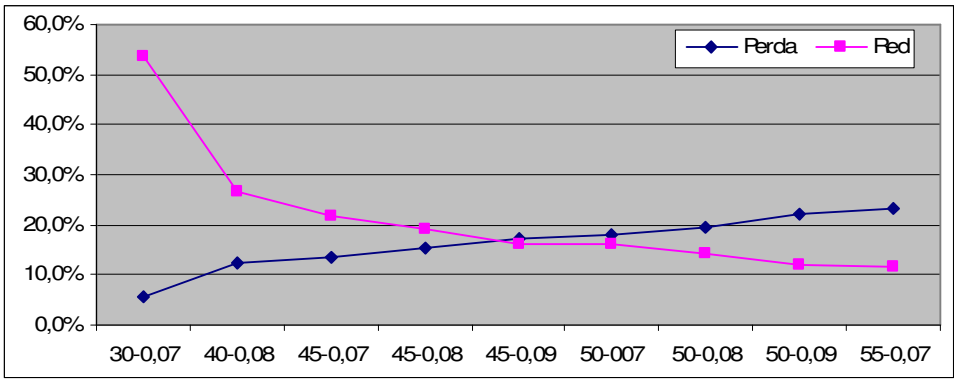


Gráfico 5 - Métodos de histograma e diferença *pixel*-a-*pixel*

Fonte: elaborado pelo autor (2008).

Na combinação desses dois métodos já é possível constatar o cruzamento das linhas de perda e redundância é melhor do que qualquer um dos métodos isolados. Ou seja, há simultaneamente valores mais baixos de perda e redundância como se deseja.

6.1.3.2 Combinação dos métodos de histograma e diferença *pixel*-a-*pixel* IQ

A combinação dos métodos de histograma e *pixel*-a-*pixel* "IQ", também apresentou resultados bem melhores que qualquer um dos seus componentes isolados. As desvantagens são herdadas dos seus componentes básicos analisados, onde há uma redundância excessiva sobre os efeitos de dissolução. Outra desvantagem ocorre em cenas em P&B, pois apenas o primeiro quadro-chave é capturado e mais nenhum outro enquanto a cena não volta a apresentar cor. De fato, esse problema é esperado, já que os canais "I" e "Q" são justamente canais de crominância enquanto que o canal "Y" cuja finalidade é carregar a luminância fora desprezado. Como ponto forte, pode-se destacar uma maior robustez em cenas onde há grande variação de luminosidade (cenas com *flashes*). Essa robustez é interessante para vídeos jornalísticos, pois, este tipo de ocorrência é relativamente comum (por exemplo, em entrevistas coletivas de pessoas públicas). A explicação para o fato é que, no caso dos *flashes*, há variação basicamente no canal de luminância "Y", justamente aquele descartado no método.

A

Tabela 9 mostra os testes feitos com alteração nos dois limiares. O primeiro se refere à distância *pixel*-a-*pixel* e o segundo, à dos histogramas "IQ".

Tabela 9 - Desempenho do método de comparação de histograma e *pixel*-a-*pixel* IQ

| Limiares | KF/S | Perda | Red. | TotP | TotR | Total |
|---|---|---|---|---|---|---|
| 10-0,22 | 1,53 | 10,9% | 30,9% | 194 | 177 | 984 |
| 10-0,23 | 1,42 | 16,7% | 18,2% | 217 | 144 | 832 |
| 11-0,20 | 1,5 | 17,8% | 16,7% | 181 | 148 | 879 |
| 11-0,21 | 1,45 | 17,9% | 15,0% | 199 | 137 | 846 |
| 11-0,23 | 1,32 | 18,5% | 14,1% | 235 | 106 | 773 |
| 12-0,22 | 1,24 | 18,7% | 16,9% | 264 | 87 | 728 |
| 10,5-0,21 | 1,54 | 19,1% | 18,5% | 184 | 175 | 901 |
| 10,75-0,2 | 1,52 | 19,5% | 13,7% | 179 | 157 | 891 |
| 10,75-0,21 | 1,47 | 20,2% | 12,7% | 194 | 145 | 860 |
| 09-0,19 | 1,92 | 21,3% | 14,6% | 107 | 311 | 1126 |
| 10,9-0,19 | 1,58 | 23,2% | 10,2% | 169 | 181 | 928 |
| 10,75-0,18 | 1,65 | 26,1% | 8,0% | 155 | 203 | 964 |

Fonte: elaborada pelo autor (2008).

O Gráfico 6 mostra as curvas resultantes para a relação entre perda e redundância na utilização dos métodos integrados. O ponto de intersecção mais baixo entre as curvas (em torno de 18,5%) ocorre nas proximidades dos limiares iguais a 11 e 0,20 para a distância *pixel*-a-*pixel* e para a distância de histogramas, respectivamente.

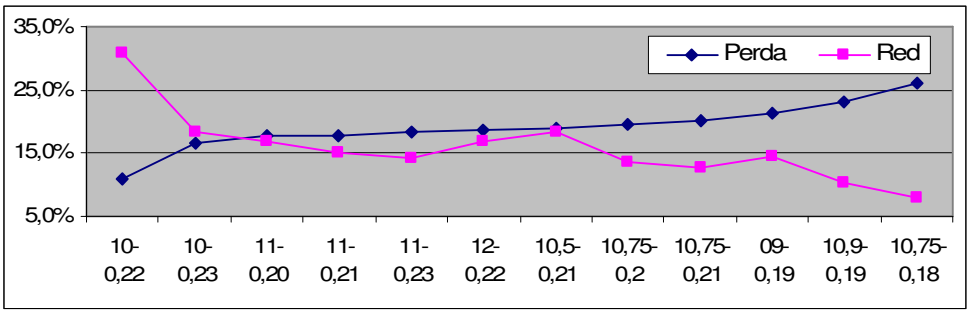


Gráfico 6 - Desempenho do método de comparação de histograma e *pixel* a *pixel* IQ

Fonte: elaborado pelo autor (2008).

6.1.3.3 Combinação dos quatro métodos básicos

Com o objetivo de se obter um melhor desempenho em termos de perda e redundância na extração de quadros-chave, foi proposta e avaliada a combinação de todos os métodos básicos apresentados. A combinação dos métodos exige a definição de quatro limiares, um para cada método básico. As definições dos limiares foram baseadas em heurística. Inicialmente, foram estabelecidos pisos para cada limiar, com base nas avaliações individuais de cada método. Em seguida, foi feita

uma avaliação do desempenho conjunto com esses valores mínimos de limiar. Após a geração do sumário do vídeo, por inspeção visual, buscou-se determinar qual dos métodos estava relacionado à geração excessiva de quadros (uma vez que se trabalhava com limiares muito baixos). O método identificado deveria ter o seu limiar aumentado a fim de minimizar o problema. De maneira similar, os limiares adequados necessitam de ajuste quando o sumário passa a ter problemas de perda excessiva de quadros. Por exemplo, se o sumário apresentar grande redundância em efeitos de *fading*, apenas o limiar relacionado com a comparação de histogramas em RGB é alterado, afim de, em seguida, um novo teste ser realizado. A Tabela 10 mostra os resultados obtidos com a combinação dos quatro métodos. Os valores dos limiares são, respectivamente, o da diferença *pixel*-a-*pixel* em RGB, diferença de histograma em RGB, diferença *pixel*-a-*pixel* em IQ e, por último, a diferença de histograma em IQ.

Tabela 10 – Combinação dos quarto métodos básicos.

| Limiares | KF/S | Perda | Red. | TotP | TotR | Total |
|---|---|---|---|---|---|---|
| 30-0,05-7-0,10 | 2,6 | 5,9% | 56,3% | 51 | 649 | 1521 |
| 38-0,06-7-0,18 | 1,69 | 10,0% | 16,8% | 116 | 202 | 986 |
| 40-0,06-7-0,18 | 1,65 | 10,7% | 15,8% | 124 | 187 | 963 |
| 40-0,06-6-0,20 | 1,59 | 11,2% | 13,5% | 135 | 161 | 926 |
| 35-0,06-8-0,18 | 1,67 | 11,4% | 16,7% | 120 | 195 | 977 |
| 40-0,06-7-0,20 | 1,54 | 11,6% | 12,6% | 140 | 145 | 902 |
| 40-0,05-7-0,20 | 1,55 | 12,1% | 12,5% | 142 | 148 | 906 |
| 42-0,06-7-0,18 | 1,59 | 12,4% | 14,1% | 138 | 165 | 926 |
| 30-0,07-8-0,18 | 1,66 | 12,7% | 18,2% | 139 | 206 | 972 |
| 40-0,06-7-0,205 | 1,49 | 14,0% | 11,9% | 164 | 139 | 872 |
| 40-0,06-8-0,20 | 1,47 | 14,3% | 11,3% | 167 | 129 | 859 |
| 40-0,06-7-0,21 | 1,45 | 15,2% | 11,3% | 178 | 130 | 849 |
| 40-0,07-7-0,20 | 1,46 | 15,6% | 11,5% | 191 | 134 | 853 |
| 35-0,06-10-0,18 | 1,5 | 15,8% | 12,5% | 166 | 142 | 878 |

Fonte: elaborada pelo autor (2008).

A Tabela 10 traz as principais avaliações de sumários gerados. Embora o ajuste simultâneo de mais de um limiar apareça nessa tabela, isso não ocorre na prática. Os ajustes foram feitos de maneira independente para cada limiar, dado que pequenas variações dos limiares podiam causar efeitos indesejáveis no resultado global da extração do sumário. A avaliação dos métodos básicos isoladamente permitiu a definição de um intervalo adequado para o ajuste de cada limiar, limitando o conjunto das possíveis configurações a serem avaliadas. O Gráfico 7 mostra o comportamento das curvas de perda e redundância com os ajustes dessas quatro variáveis.

Dentre todos os métodos testados, individual e em conjunto, esse último apresentou os melhores resultados de relação perda *versus* redundância, sendo escolhidos os valores 40 para a diferença *pixel*-a-*pixel* RGB, 0,05 para diferença de histogramas RGB, 7 para diferença *pixel*-a-*pixel* IQ e 0,20 para diferença de histograma IQ.

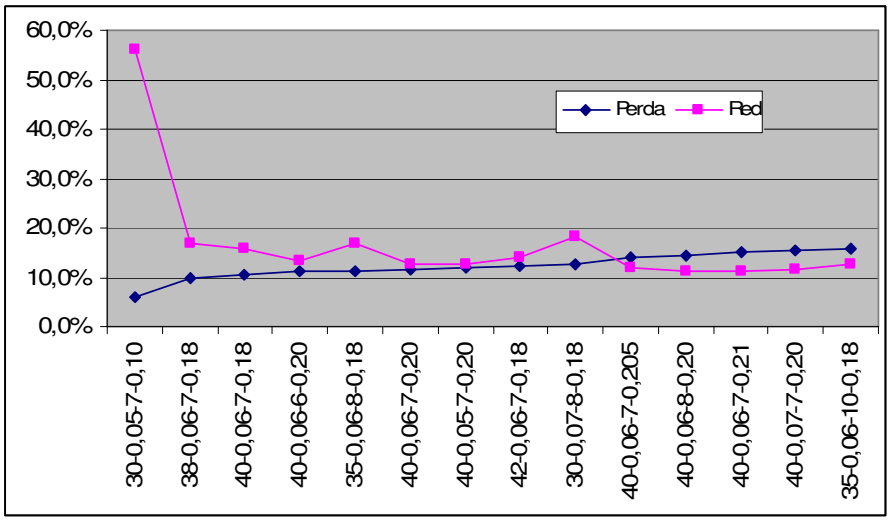


Gráfico 7 - Combinação dos quatro métodos básicos.

Fonte: elaborado pelo autor (2008).

A Tabela 11 mostra a quantidade de quadros-chave capturados de cada vídeo e o percentual em relação ao total de quadros.

Tabela 11 – Percentual de quadros-chave capturados

| Data da Edição | Total de Quadros | Total de Quadros Chave | Percentual de Quadros Chave | Data da Edição | Total de Quadros | Total de Quadros Chave | Percentual de Quadros Chave |
|---|---|---|---|---|---|---|---|
| 26/9/2007 | 79432 | 886 | 1,12% | 6/12/2007 | 81113 | 1017 | 1,25% |
| 27/9/2007 | 63646 | 830 | 1,30% | 7/12/2007 | 25722 | 375 | 1,46% |
| 28/9/2007 | 74944 | 1358 | 1,81% | 18/1/2008 | 77310 | 884 | 1,14% |
| 1/10/2007 | 69122 | 822 | 1,19% | 19/1/2008 | 75820 | 1050 | 1,38% |
| 31/10/2007 | 72122 | 978 | 1,36% | 21/1/2008 | 74091 | 824 | 1,11% |
| 2/11/2007 | 71123 | 1056 | 1,48% | 22/1/2008 | 74688 | 887 | 1,19% |
| 5/11/2007 | 75785 | 956 | 1,26% | 23/1/2008 | 54012 | 631 | 1,17% |
| 13/11/2007 | 81797 | 1158 | 1,42% | 24/1/2008 | 84903 | 996 | 1,17% |
| 14/11/2007 | 63006 | 898 | 1,43% | 25/1/2008 | 82461 | 895 | 1,09% |
| 15/11/2007 | 61106 | 890 | 1,46% | 26/1/2008 | 84271 | 1146 | 1,36% |
| 16/11/2007 | 69831 | 929 | 1,33% | 28/1/2008 | 78885 | 879 | 1,11% |
| 20/11/2007 | 77333 | 876 | 1,13% | 29/1/2008 | 82072 | 1031 | 1,26% |
| 21/11/2007 | 55921 | 704 | 1,26% | 30/1/2008 | 72518 | 902 | 1,24% |
| 22/11/2007 | 82755 | 1043 | 1,26% | 1/2/2008 | 72789 | 892 | 1,23% |
| 23/11/2007 | 77797 | 1022 | 1,31% | 2/2/2008 | 76845 | 1126 | 1,47% |
| 26/11/2007 | 82771 | 917 | 1,11% | 4/2/2008 | 21474 | 321 | 1,49% |
| 27/11/2007 | 81666 | 969 | 1,19% | 6/2/2008 | 70444 | 730 | 1,04% |
| 28/11/2007 | 67642 | 882 | 1,30% | 7/2/2008 | 84367 | 965 | 1,14% |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 29/11/2007 | 65674 | 793 | 1,21% | 8/2/2008 | 81964 | 880 | 1,07% |
| 30/11/2007 | 77829 | 1092 | 1,40% | 9/2/2008 | 78155 | 983 | 1,26% |
| 3/12/2007 | 70688 | 967 | 1,37% | 11/2/2008 | 80148 | 808 | 1,01% |
| 4/12/2007 | 81169 | 1037 | 1,28% | 12/2/2008 | 82944 | 758 | 0,91% |
| 5/12/2007 | 79684 | 1094 | 1,37% | 13/2/2008 | 65939 | 759 | 1,15% |

Fonte: elaborada pelo autor (2008).

### 6.1.4 Resultado da captura de quadros-chave

Uma vez definidos os melhores limiares dos quatro métodos em conjunto, foram processados todos os vídeos jornalísticos da base. Esses vídeos formam um conjunto de 46 edições do Jornal Nacional, totalizando pouco mais que 34 horas de vídeo, inclusos intervalos comerciais. Dessa base, o total de 3.420.835 quadros fora reduzido para 42.755 quadros-chave, usados para a busca tanto baseada em características estatísticas quanto baseada na assinatura *wavelet*. Isso representa uma redução na quantidade total de quadros para em média 1,25% da base total.

Embora os melhores valores de limiares para sumarização, obtidos nos testes possam variar de um vídeo para outro, uma vez estabelecidos estes limiares com o vídeo de teste, esses valores foram replicados para as demais edições da base, isso por se tratar de uma mesma categoria de vídeo (tele-jornais). Para outras categorias, o estudo na variação dos limiares pode ser feito direto na combinação dos quatro métodos, já eliminando os demais métodos por não terem apresentado resultados consideráveis.

## 6.2 AVALIAÇÃO DOS MÉTODOS DE DETECÇÃO DE CORTE DE TOMADAS

As seções seguintes mostram os resultados obtidos pelos três métodos de comparação de quadros descritos em 5.1.1.2. Para avaliação de performance na detecção de cortes de tomada, esse trabalho utiliza as métricas tradicionalmente abordadas em outros trabalhos (GUIMARÃES, 2003; SIMÕES, 2004; MANZATO; GOULARTE, 2007; CERNEKOVA.; NIKOU; PITAS, 2002; TAHAGHOGHI *et al.*, 2005), que são a comparação de histogramas das imagens e a comparação *pixel*-a-*pixel* entre quadros. São avaliadas as tomadas corretamente detectadas, as tomadas não detectadas e as tomadas erroneamente detectadas pelos métodos de comparação de quadros. Formalmente, são definidas as seguintes medidas de desempenho:

a) **verdadeiro positivo** *vp* - É um evento de corte de tomada real que tenha sido detectado pelo método automático;

b) **falso positivo** *fp* - É um evento de corte de tomada detectado erroneamente pelo método, mas na realidade esse corte não existe;
c) **falso negativo** *fn* - É perda da detecção pelo método de um corte de tomada.

A avaliação de performance é dada pelas quatro medidas: precisão, revocação, erro e qualidade (TAHAGHOGHI *et al*., 2005).

**Precisão** *p* – a precisão indica a fração de cortes detectados que combinam com os verdadeiros positivos.

$$p = \frac{nvp}{nvp + nfp}$$

**Revocação** *r* – a revocação mede a fração de todos os cortes conhecidos que são corretamente detectados.

$$r = \frac{nvp}{nvp + nfn}$$

**Erro de detecção** *e* – é a taxa de erro obtida pelo algoritmo

$$e = \frac{nfp}{nvp + nfn}$$

**Qualidade** *q* - É uma medida que combina a revocação e precisão, favorecendo a revocação.

$$q = \frac{r}{3} * \left(4 - \left(\frac{1}{p}\right)\right)$$

Nas equações de avaliação de desempenho, *nvp* denota a quantidade total de verdadeiros positivos, *nfp* denota a quantidade total de falsos positivos e *nfn* o número total de falsos negativos.

### 6.2.1 Resultados da detecção automática de corte de tomada

Os testes de desempenho dos métodos automáticos de cortes de tomada utilizaram como entrada de dados a edição do dia 26 de Setembro de 2007 do Jornal Nacional. Essa amostra tem duração de 44 minutos e 30 segundos, contendo também os intervalos comerciais. Foram identificados manualmente, 582 cortes de tomada, incluído 28 transições graduais. Destas transições, foram encontrados efeitos de *wipe* (4), *fading* (10) e principalmente dissoluções (14). Com isso, já se sabe *a priori*, que 4,8% dos cortes não são detectáveis pelos métodos aplicados.

O primeiro método avaliado foi a distância entre quadros *pixel*-a-*pixel* nos canais de cor RGB, conforme 2.7.1.1. Foi calculada a distância média, nos três canais. A Tabela 12 mostra os testes realizados com diferentes limiares e seus respectivos desempenhos.

Tabela 12 – Desempenho do método de corte de tomada *pixel* a *pixel*

| Limiar | *Qualidade* | *Erro* | *Revocação* | *Precisão* | *nfn* | *nfp* | *nvp* |
|---|---|---|---|---|---|---|---|
| 41 | 77% | 28% | 86% | 76% | 81 | 162 | 501 |
| 42 | 78% | 25% | 86% | 77% | 82 | 146 | 500 |
| 43 | 77% | 24% | 85% | 78% | 86 | 138 | 496 |
| 44 | 77% | 22% | 84% | 80% | 93 | 126 | 489 |
| 45 | 76% | 20% | 83% | 80% | 101 | 118 | 481 |
| 46 | 76% | 19% | 82% | 81% | 105 | 109 | 477 |
| 47 | 75% | 17% | 81% | 83% | 110 | 99 | 472 |
| 48 | 75% | 16% | 80% | 84% | 116 | 92 | 466 |
| 49 | 73% | 15% | 78% | 84% | 126 | 87 | 456 |
| 50 | 72% | 14% | 77% | 84% | 135 | 82 | 447 |
| 51 | 71% | 13% | 76% | 85% | 142 | 76 | 440 |

Fonte: elaborada pelo autor (2008).

O Gráfico 8 mostra a evolução das medidas de desempenho apresentadas. Com esse gráfico, é escolhido como melhor limiar para o método, o ponto em que a linha da precisão corta a linha da revocação. Nesse caso, o melhor limiar para corte de tomadas do método *pixel*-a-*pixel* é o limiar igual a 46.

Dentre os três métodos analisados, a diferença *pixel*-a-*pixel* apresentou os melhores resultados, com precisão de 81% e revocação de 82% no limiar igual a 46.

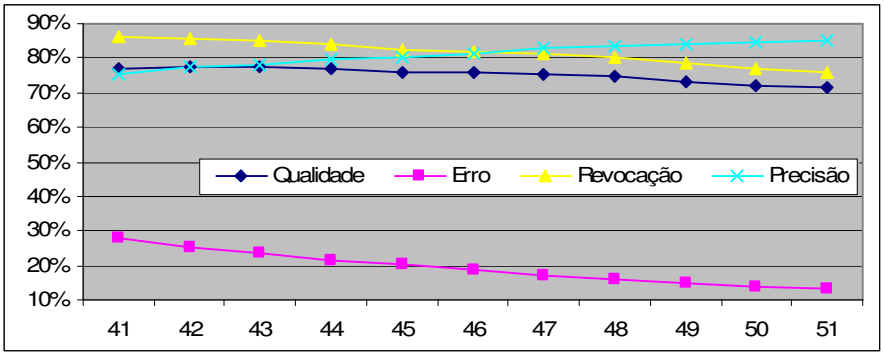


Gráfico 8 - Desempenho do método de corte de tomada *pixel*-a-*pixel*

Fonte: elaborado pelo autor (2008).

Outro método importante para detecção de corte de tomadas é a comparação dos histogramas dos quadros. No presente trabalho, foram comparados os quadros no espaço de cor RGB, usando a métrica de comparação descrita no item 2.7.1.2. A Tabela 13 mostra os testes realizados com diferentes limiares e seus respectivos desempenhos no método de comparação de histogramas.

Tabela 13 - Desempenho do método de corte de tomada por diferença de histograma

| Limiar | *Qualidade* | *Erro* | *Revocação* | *Precisão* | *nfn* | *nfp* | *nvp* |
|---|---|---|---|---|---|---|---|
| 0.08 | 58% | 77% | 83% | 52% | 98 | 447 | 484 |
| 0.09 | 56% | 67% | 78% | 54% | 126 | 391 | 456 |
| 0.10 | 53% | 58% | 72% | 55% | 161 | 339 | 421 |
| 0.11 | 51% | 48% | 67% | 58% | 191 | 279 | 391 |
| 0.12 | 47% | 43% | 62% | 59% | 223 | 251 | 359 |
| 0.13 | 43% | 39% | 56% | 59% | 257 | 229 | 325 |
| 0.14 | 39% | 37% | 51% | 58% | 285 | 213 | 297 |
| 0.15 | 37% | 34% | 48% | 59% | 300 | 197 | 282 |
| 0.16 | 34% | 32% | 45% | 58% | 320 | 186 | 262 |
| 0.17 | 32% | 30% | 41% | 58% | 341 | 172 | 241 |
| 0.18 | 30% | 27% | 39% | 59% | 355 | 157 | 227 |

Fonte: elaborada pelo autor (2008).

Embora fossem esperados resultados melhores, o método de comparação de histogramas apresentou os piores resultados dentre os três avaliados. A melhor relação de revocação por precisão foi obtida com limiar igual a 0,12.

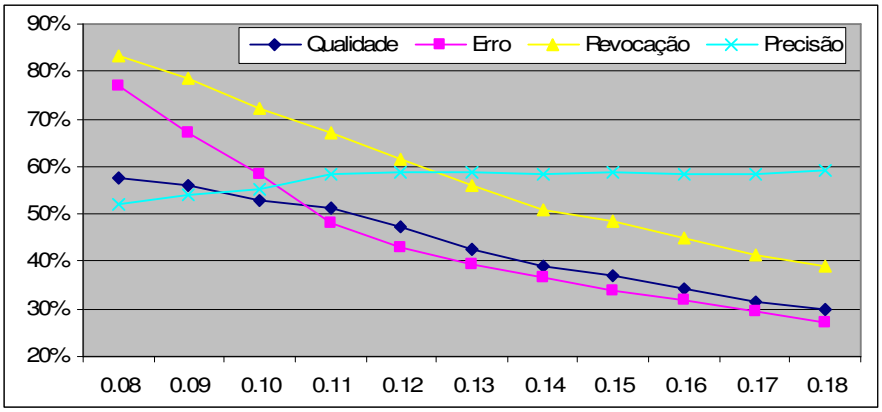


Gráfico 9 – Desempenho do método de corte de tomada por diferença de histograma

Fonte: elaborado pelo autor (2008).

O terceiro e último método avaliado foi a comparação de quadros adjacentes através da assinatura *wavelet*. A métrica é descrita no item 4.8.2 com a eq. 11 com a tabela de pesos de comparação de assinatura da " busca por exemplo". A Tabela 14 mostra os testes realizados com diferentes limiares e seus respectivos desempenhos no método de comparação de assinaturas. Como pode ser observado, este método apresenta um desempenho ligeiramente melhor que a comparação de histogramas, contudo um desempenho menor que a comparação *pixel*-a-*pixel*.

Tabela 14 - Desempenho do método de corte de tomada por assinatura *wavelet*

| Limiar | *Qualidade* | *Erro* | *Revocação* | *Precisão* | *nfn* | *nfp* | *nvp* |
|---|---|---|---|---|---|---|---|
| 130 | 66% | 46% | 81% | 64% | 109 | 269 | 473 |
| 140 | 65% | 43% | 79% | 65% | 121 | 248 | 461 |
| 150 | 65% | 37% | 77% | 68% | 132 | 214 | 450 |
| 160 | 64% | 34% | 75% | 69% | 145 | 195 | 437 |
| 170 | 61% | 32% | 72% | 69% | 165 | 184 | 417 |
| 180 | 60% | 29% | 69% | 71% | 178 | 167 | 404 |
| 190 | 59% | 26% | 68% | 72% | 188 | 154 | 394 |
| 200 | 59% | 24% | 67% | 74% | 193 | 139 | 389 |
| 210 | 58% | 22% | 65% | 75% | 201 | 126 | 381 |
| 220 | 57% | 21% | 64% | 76% | 210 | 120 | 372 |
| 230 | 57% | 19% | 63% | 77% | 216 | 108 | 366 |

Fonte: elaborada pelo autor (2008).

O melhor limiar para essa métrica foi de 180, com o cruzamento da precisão com a revocação conforme mostrado no Gráfico 10.

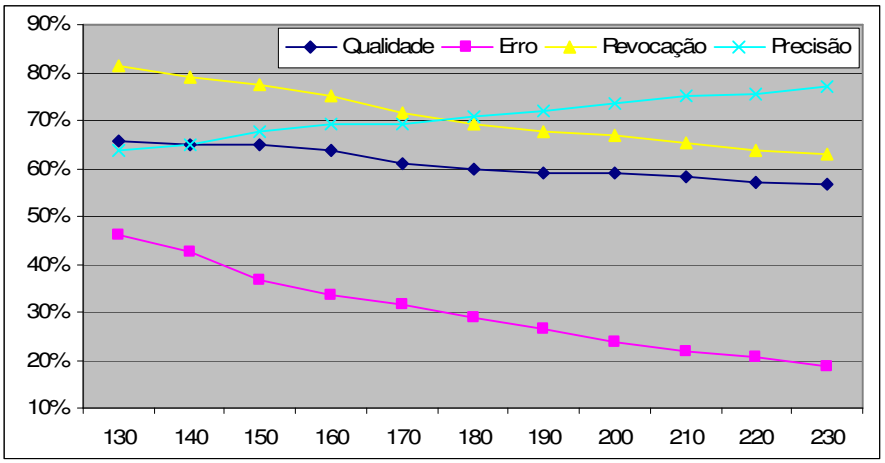


Gráfico 10 - Desempenho do método de corte de tomada por assinatura de *wavelet*

Fonte: elaborado pelo autor (2008).

Embora fossem esperados resultados melhores na comparação de quadros por assinatura de *wavelet*, os resultados de comparação *pixel*-a-*pixel,* apesar de simples, foram os mais eficientes.

As seções seguintes apresentam os resultados de desempenho nos diversos métodos de busca de quadros de vídeos baseado em similaridade visual, sendo esse, o principal objetivo do presente trabalho. Inicialmente, são descritas as métricas utilizadas para avaliação de performance e em seguida os resultados propriamente ditos. Além dos métodos de busca, os resultados são afetados pelos filtros aplicados antes da classificação dos resultados descrito no item 5.2.

## 6.3 AVALIAÇÃO DA RECUPERAÇÃO DE QUADROS POR SIMILARIDADE VISUAL

A avaliação de desempenho das abordagens de CBIR não é um trabalho trivial por dois motivos: Primeiro, não há uma padronização das métricas utilizadas por cada abordagem; segundo, não há uma base padrão para avaliação do resultados das buscas (FENG; SIU; ZHANG, 2003).

Para avaliar o desempenho das buscas de quadros, são utilizadas duas medidas tradicionais, emprestadas da recuperação de informação de modo genérico: são as medidas de revocação e a precisão (FENG; SIU; ZHANG, 2003), que já foram empregadas anteriormente, neste trabalho, na avaliação de corte de tomadas. Especificamente na avaliação de desempenho de busca de quadros, $\mu$ denota o número total de quadros relevantes recuperados, $\lambda$ o número total de quadros recuperados e $\vartheta$ número total de quadros relevantes. As medidas de precisão *p* e revocação *r* são:

$$p = \frac{\mu}{\lambda}, r = \frac{\mu}{\vartheta}$$

O número total de quadros relevantes, $\vartheta$, pode variar muito de acordo com o tipo de quadro procurado. Por exemplo, a busca de um quadro atípico, terá poucas ocorrências em toda base de vídeos, entretanto, a busca de um quadro que faz parte de uma vinheta, pode se repetir muitas vezes em toda base, pois cada vez que a vinheta é exibida, uma nova ocorrência desse quadro deverá integrar a base.

A avaliação de desempenho da busca de cada quadro é feita em duas etapas; na primeira etapa são contadas manualmente as ocorrências dos quadros similares, ou seja, definida uma busca, quantos quadros espera-se que sejam retornados. A segunda etapa é a comparação da quantidade retornada pelo método automático contra o método manual. Essa segunda etapa ainda segue alguns critérios importantes relacionados com a posição do quadro na classificação de similaridade, pois não basta que o quadro seja retornado, o mesmo precisa estar bem classificado. Essa posição de classificação do quadro está diretamente relacionada com a equação da precisão. O ideal é que os quadros alvo sejam retornados e classificados entre os primeiros resultados, o que na prática nem sempre ocorre, assim foi admitida uma certa faixa de tolerância para a posição máxima admitida na classificação. A faixa de tolerância é relativa, pois, depende da quantidade de quadros alvos esperados. Sendo assim, cada quadro é avaliado

quatro vezes em relação às suas medidas de precisão e de revocação. O motivo dessa avaliação em quatro etapas foi dar uma maior margem de tolerância na classificação da busca dos quadros. Como conseqüência, observa-se que quando é aumentado o conjunto de amostras classificadas, surge um aumento na revocação e uma queda na precisão.

As quatro avaliações seguem o seguinte critério: para a precisão $p_i$, $\lambda$ varia $\lambda = i\vartheta$, sendo que $i \in \{1,2,3,4\}$. Na prática, isto quer dizer que para a primeira busca, são avaliados os quadros classificados até a posição $\vartheta$; na segunda busca são avaliados os resultados até $2\vartheta$; na terceira até $3\vartheta$ e finalmente, na quarta e última busca, são avaliados os resultados de precisão e revocação até $4\vartheta$.

Embora seja claro que nas buscas feitas “em produção”, a quantidade de quadros considerados corretos não seja conhecida *a priori* pelo usuário, o fato de avaliar-se até no máximo quatro vezes a quantidade manual é uma métrica formal para a avaliação de desempenho no presente trabalho. Na prática, um usuário real da ferramenta pode navegar por um conjunto realmente muito maior de quadros retornados e quanto maior esse conjunto mais provável será a obtenção de uma revocação de 100%, ou muito próxima disso.

### 6.3.1 Conjunto de quadros pesquisados

As buscas foram feitas num conjunto de 46 vídeos ou edições do Jornal Nacional conforme Tabela 11, totalizando pouco mais que 34 horas de tele-jornal inclusos intervalos comerciais. Desse conjunto de vídeos, o total de 3.420.835 quadros fora reduzido para 42.755 quadros-chave através dos métodos de sumarização apresentados. É nesse conjunto de quadros-chave reduzido que são realizadas as buscas com base em características estatísticas e pela assinatura *wavelet*.

### 6.3.2 Resultados da busca de quadros mais similares.

Para esse experimento, foram considerados os métodos de busca baseado em estatística e dois métodos baseados em *wavelet* (“busca por exemplo” e “ busca por rascunho”). No contexto do presente trabalho, foram testadas 21 imagens em cada método. Sendo que essas imagens foram quadros selecionados da base de quadros indexados, que já se sabia repetirem-se algumas vezes em toda base de vídeos. A depender do quadro, alguns se repetiam mais vezes enquanto que outros

menos. Por exemplo, um quadro retirado da vinheta de abertura de intervalo comercial do Jornal Nacional, na base de quadros do presente trabalho, repete-se 187 vezes, enquanto que; um quadro retirado da vinheta da matéria da previsão do tempo 45 vezes (não foram 46 vezes, porque uma das edições da base está incompleta). As outras imagens testadas foram basicamente logomarcas ou quadros que se repetem nos comerciais da base.

O método de busca baseado nos coeficientes de *wavelet* "por rascunho" apresentou os melhores resultados, em seguida a busca baseada em estatística e finalmente a busca baseada em *wavelet* "por exemplo", com os piores resultados. É importante destacar que no método utilizado na apresentação dos resultados do Gráfico 13 foi de assinatura de *wavelet* "por rascunho", mas, as imagens de busca *Q* usadas foram exemplos. Apesar dos métodos e a imagem de busca discordarem conceitualmente não há nenhum impedimento técnico na realização desse experimento. Os testes que utilizam imagem de busca *Q* com rascunhos no método de assinatura de *wavelet* "por rascunho", são apresentados no item 6.3.4.

O Gráfico 11 mostra a evolução média dos 21 testes com as medidas de revocação e precisão das buscas. Este gráfico representa o método de busca baseado em características estatísticas descrita no item 5.1.1.5.

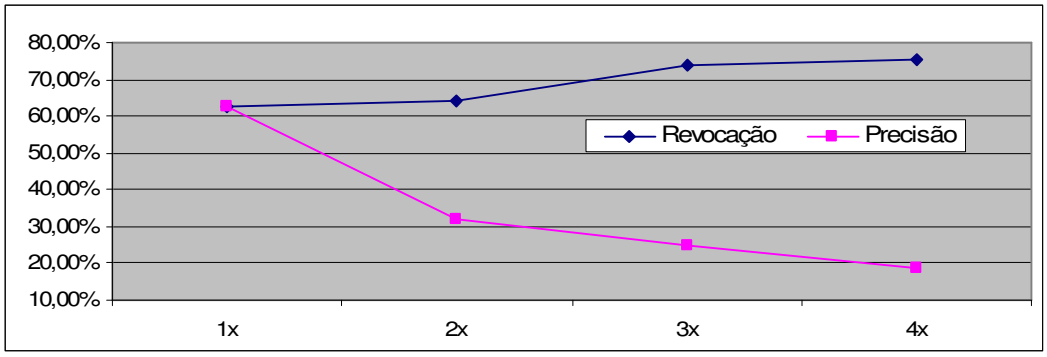


Gráfico 11 - Revocação e precisão da busca de quadros mais similares por estatística

Fonte: elaborada pelo autor (2008).

Conforme mostrado no Gráfico 11, o método baseado em estatística apresentou pouco mais de 60% de precisão e revocação nos primeiros lugares classificados. Aumentando-se a quantidade de quadros retornados na busca é natural uma queda na precisão e um aumento na revocação. Esse aumento foi de pouco mais de 13 pontos percentuais para a revocação, considerando o limite de

quadros de retorno de 4 vezes a quantidade esperada na contagem manual dos testes.

Foi observado que a busca baseada em estatística, utilizando-se um quadro de exemplo que se saiba já estar contido na base, realmente tende a trazer nos primeiros lugares, os quadros com similaridade visual semântica e, nos demais lugares, quadros apenas com similaridade visual. Obviamente que se o quadro fornecido como exemplo não consta na base, os quadros retornados tendem a ser apenas “visualmente similares” ao quadro de busca (isto é, sem nenhum tipo de correlação semântica). Observa-se que o método estatístico é bastante sensível a pequenas variações de brilho e saturação. Com isso, em algumas situações, quadros semanticamente similares acabaram não sendo bem classificados devido a estes tipos de mudanças no conteúdo das edições do Jornal Nacional armazenadas na base. A Figura 36 mostra um exemplo de busca baseada em estatística.

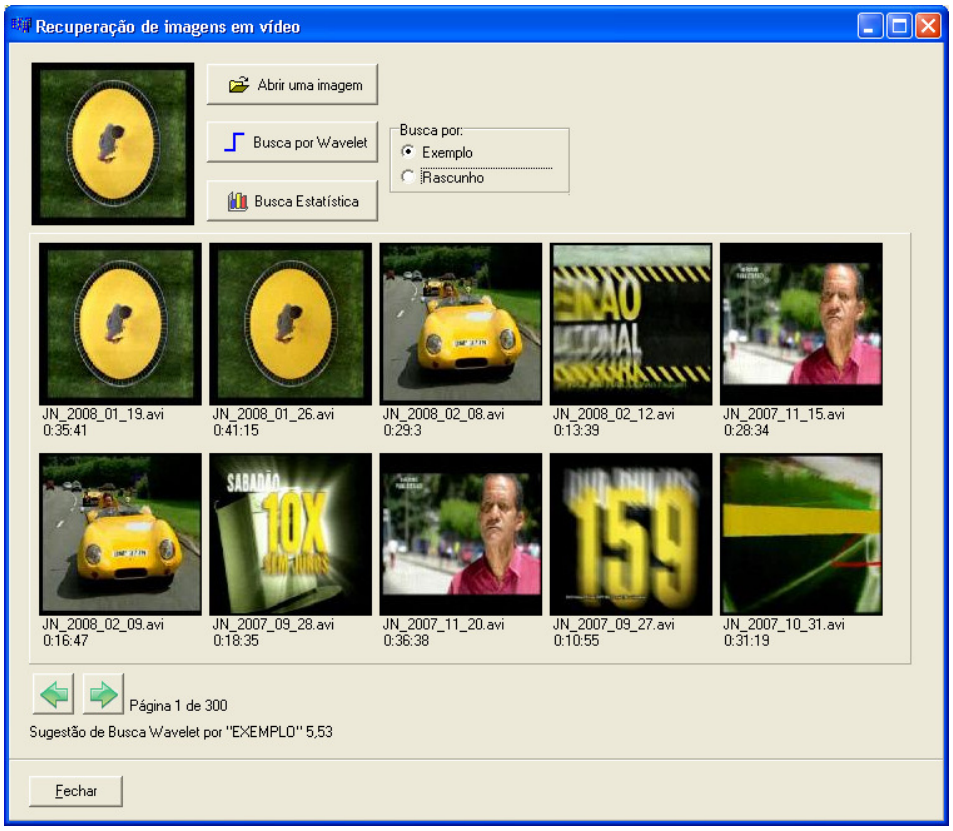


Figura 36 – Exemplo de busca baseada em estatística

Fonte: elaborada pelo autor (2008).

No método seguinte, apresentado no Gráfico 12, são mostrados os resultados de revocação e precisão obtidos com a busca baseada em *wavelet* “por exemplo”.

Os resultados mostram-se menos favoráveis que os obtidos com o método estatístico e o de *wavelet* "por rascunho".

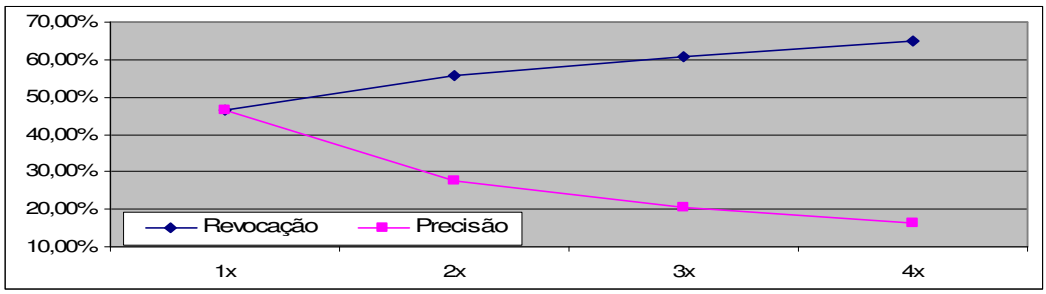


Gráfico 12 - Revocação e precisão da busca de quadros por assinatura *wavelet* " busca por exemplo".

Fonte: elaborado pelo autor (2008).

Uma vez que esse método, proposto originalmente pelos autores Jacobs, Finkelstein e Salesin (1995), apresenta um peso considerável na média de *pixels* dos canais de cor, ou seja, nos coeficientes de média, esse método também acabou se mostrando um tanto vulnerável a alterações em intensidade de brilho dos diferentes vídeos da base do presente trabalho.

A seguir, no Gráfico 13, são apresentados os melhores resultados dentre os métodos testados para as buscas de quadros mais similares, com uma larga vantagem percentual sobre os demais métodos. Na busca por assinatura *wavelet* com os pesos da tabela de rascunho, houve uma revocação maior que 90% já nos primeiros resultados dos quadros de busca, com uma pequena evolução na revocação chegando a 95,6% quando admitida uma queda na precisão.

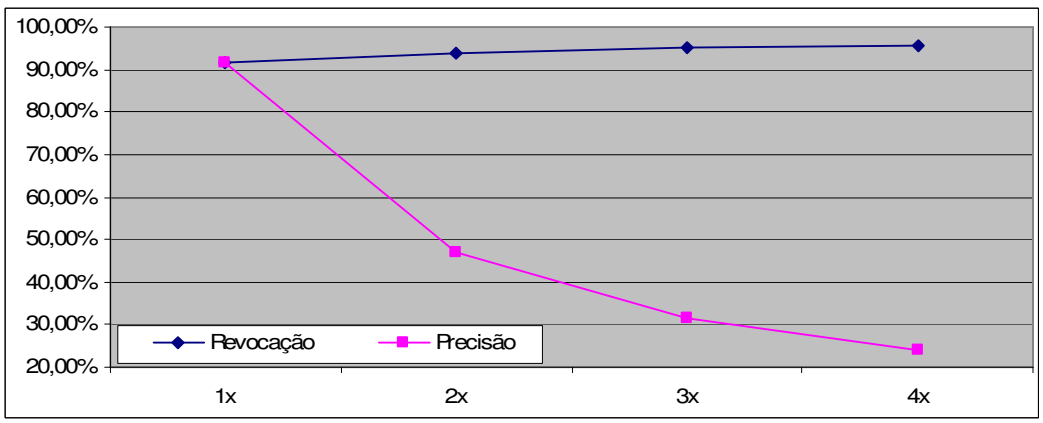


Gráfico 13 - Revocação e precisão da busca de quadros por assinatura de *wavelet* "por rascunho".

Fonte: elaborado pelo autor (2008).

No método da "busca por rascunho", variações de intensidade de luz não apresentam o mesmo impacto que nos métodos estatísticos e de assinatura de

*wavelet* da “busca por exemplo”. Outro ponto forte nesse método é o filtro com maior abertura trazendo, por conseguinte, um conjunto maior de quadros possíveis, os quais são melhor classificados com base nos coeficientes de detalhe da assinatura *wavelet*. Realmente esse método mostrou-se bastante resistente, inclusive a grandes variações de intensidade de brilho, como pode ser visto nos resultados da busca ilustrado na Figura 37. Nessa figura, é possível notar que os quadros apresentados respectivamente na oitava e nona posição apresentam uma intensa distorção na variação do brilho com relação ao quadro utilizado como exemplo de entrada para busca, entretanto, os quadros são recuperados e classificados em ordem adequada com relação à similaridade visual do quadro de busca.

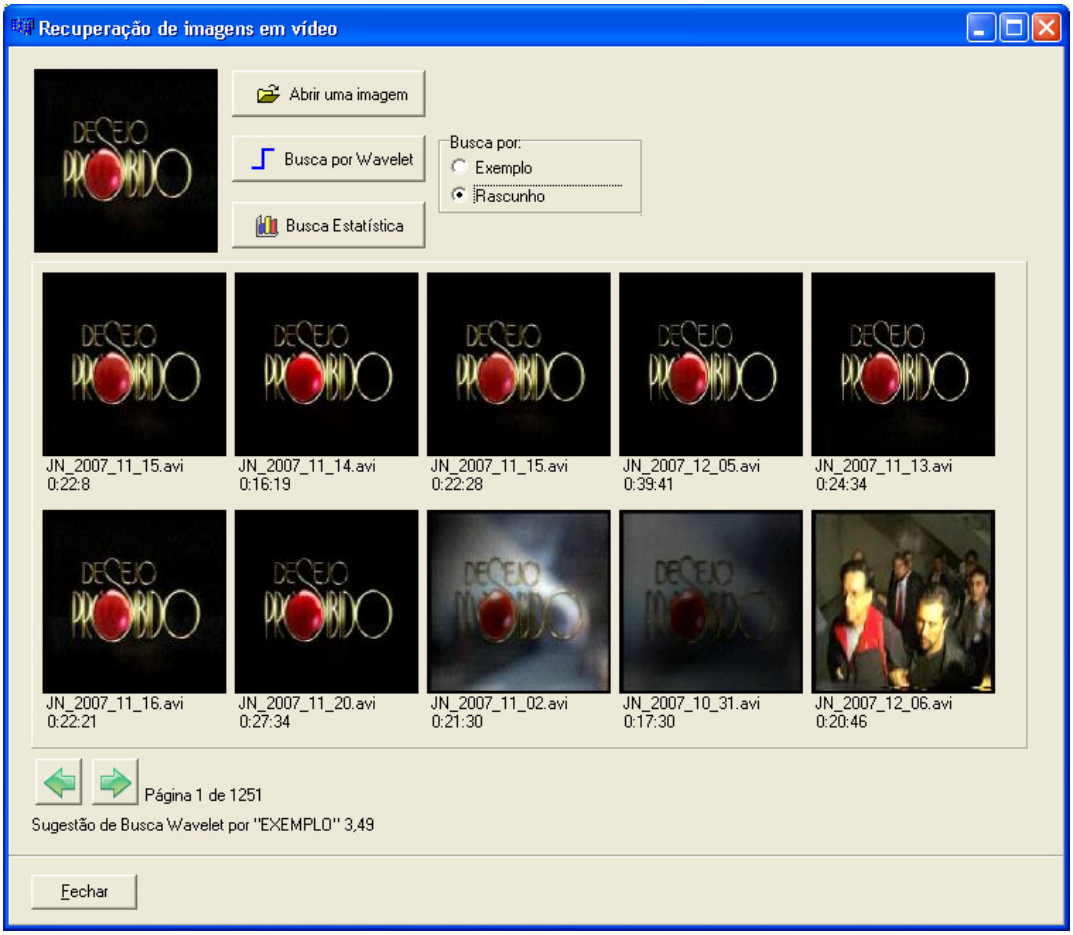


Figura 37 – Exemplo de busca baseada em *wavelet* por rascunho

Fonte: elaborada pelo autor (2008).

Embora não haja implementação de inteligência artificial, além do fato de que o sistema não tenha sido proposto nem construído com o objetivo de recuperação de quadros com conteúdo semelhante sob o ponto de vista semântico, dentre os três

métodos avaliados, esse último, de certa forma, consegue agrupar os quadros com mesmo conteúdo de forma mais adequada que os demais.

O tipo de pesquisa que apresentou os resultados menos satisfatórios foram as buscas por assinatura *wavelet* utilizando a tabela de pesos dos coeficientes para o método da "busca por exemplo" proposto pelos autores (JACOBS; FINKELSTEIN; SALESIN, 1995). Por outro lado, o mesmo método com um filtro primário mais abrangente e a tabela de pesos "por rascunho" apresenta resultados muito superiores.

### 6.3.3 Resultados da busca de quadros com tolerância a mudanças

Nessas buscas é considerada uma tolerância a mudanças conforme descrito na seção 5.3 Para esses testes foram selecionadas 25 imagens de exemplo do Jornal Nacional. Todos esses exemplos foram imagens contendo algum apresentador âncora chamando uma notícia com um fundo temático relacionado à notícia. As mudanças nas imagens selecionadas nessa base de exemplos ocorrem de diversas formas, por exemplo: na cor do terno ou o próprio apresentador e em detalhes do fundo temático à notícia. Normalmente esses fundos são formados por algum efeito de animação que constrói o tema, por isso, nem sempre o quadro-chave capturado na sumarização tem o fundo exatamente da mesma forma. Por exemplo, Na terceira imagem da primeira linha mostrada na Figura 38, o efeito de construção do fundo ainda não havia sido completamente concluído, faltando o escudo da seleção brasileira na imagem. Dessa forma, deseja-se que o sistema de busca seja tolerante o suficiente nessa carência de detalhes. A Figura 38 mostra alguns exemplos utilizados nos testes, sendo notícias sobre educação, saúde, seleção brasileira, vôlei, Fundo Monetário Internacional, notícia internacional, eleições americanas, notícias do presidente do Brasil, campeonato brasileiro e entorpecentes.

Figura 38 – Exemplos de imagens com tolerância a mudanças

Fonte: elaborada pelo autor (2008).

Entendendo dessa forma, o método escolhido por mais se adequar ao desafio foi a busca por *wavelet* com os parâmetros da busca “por rascunho”, apesar das imagens de busca serem exemplos e não rascunhos. Mais uma vez, embora haja uma alteração no conceito, não há nenhum impedimento técnico para tal.

Os resultados da busca são mostrados no Gráfico 14 e os parâmetros para avaliação de desempenho seguem os mesmos critérios dos métodos descritos anteriormente.

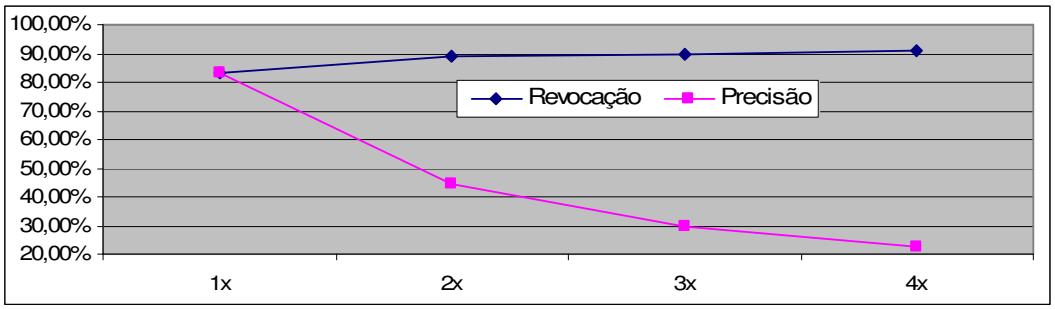


Gráfico 14 – Busca com tolerância a mudanças

Fonte: elaborado pelo autor (2008).

Os resultados foram bastante satisfatórios em diversos casos, como no exemplo de chamadas para notícias internacionais onde a bandeira do país no tema é alterada.

### 6.3.4 Busca de quadros rascunhados

Esse é um teste onde de fato são utilizadas imagens de busca rascunhadas e pesquisadas com método de assinatura de *wavelet* “por rascunho”, já que anteriormente esse método foi apresentado no presente trabalho utilizando, na entrada, imagens de exemplo. A Figura 39 mostra algumas imagens que foram rascunhadas para esses testes.

Figura 39 – Imagens rascunhadas

Fonte: elaborada pelo autor (2008).

Os rascunhos utilizados foram criados com base nos exemplos usados nos testes apresentados nos gráficos de 11 a 13. Nesses casos específicos, as imagens de entrada foram redesenhadas manualmente, configurando assim de fato uma imagem de rascunho.

Os resultados desse tipo de busca foram bastante satisfatórios quando as imagens apresentavam realmente uma boa semelhança na posição dos objetos. A baixa quantidade de cores únicas não representou um problema grave nessa metodologia. Os resultados são apresentados no Gráfico 15 e mostram que a revocação chega a atingir uma média de 90% de sucesso até as 4x primeiras colocações.

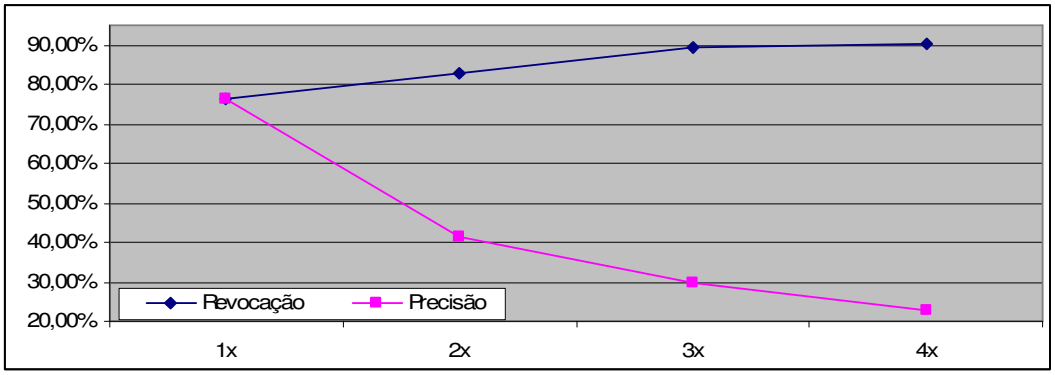


Gráfico 15 – Busca de quadros através de imagens rascunhadas

Fonte: elaborado pelo autor (2008).

O sucesso da busca está intimamente ligado a qualidade do rascunho, e nos casos em que o rascunho é feito por memória, quando o usuário cria o “rabisco” apenas com as informações de sua lembrança, a tendência é a criação de um modelo ruim tendo como resultado o insucesso das buscas, contudo, esse estudo não é o foco do presente trabalho.

### 6.3.5 Resultados da busca de quadros sorteados

Nesse tópico, uma última avaliação é proposta com o objetivo de mostrar que o ambiente apresentado neste trabalho é viável, tanto na seleção dos quadros-chave quanto na busca de quadros. Nessa avaliação, foi sorteado um conjunto de 25 quadros da edição do dia 26 de setembro de 2007 do Jornal Nacional. Esses 25 quadros foram utilizados como quadros de exemplo nos três métodos de busca propostos: estatístico, *wavelet* “por rascunho” e *wavelet* “por exemplo”. O objetivo é mostrar que, se todo o vídeo tiver quadros-chave adequados compondo um sumário

que o represente adequadamente, e se cada quadro sorteado for visualmente similar a pelo menos um quadro pertencente ao sumário, e ainda, sendo os métodos de busca propostos também adequados, espera-se, no caso ótimo, que 100% dos quadros sorteados sejam encontrados pelo ambiente proposto. A metodologia de avaliação das buscas é a mesma proposta com precisão e revocação descrita no item 6.3.

A primeira busca realizada, baseada em similaridade estatística, tem seus resultados apresentados no Gráfico 16. Esse gráfico mostra que a revocação inicial fica pouco abaixo dos 70% evoluindo para próximo de 75% ao se reduzir a precisão.

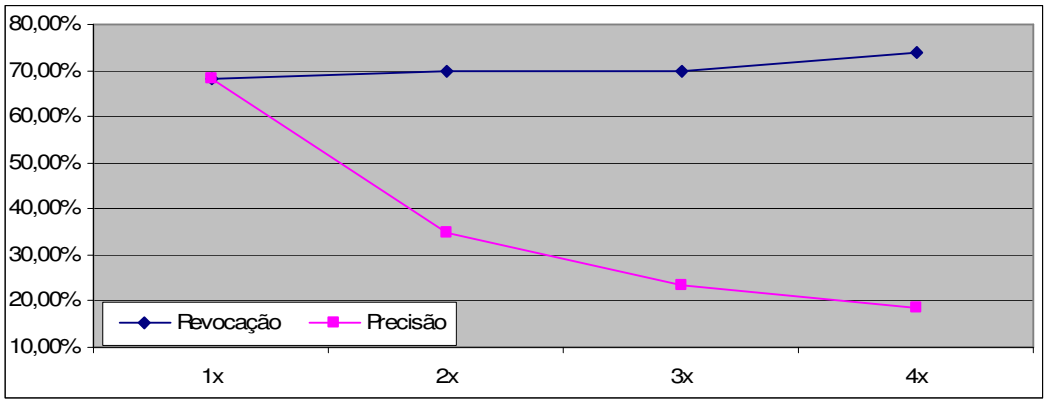


Gráfico 16 – Busca estatística de quadros sorteados

Fonte: elaborado pelo autor (2008).

O Gráfico 17, baseado no método de busca por assinatura de *wavelet* "por exemplo", apresenta resultados bastante semelhantes a busca estatística. A revocação inicial também fica abaixo dos 70%, contudo, a revocação final com a precisão em 20% fica ligeiramente superior ao método baseado em similaridade estatística.

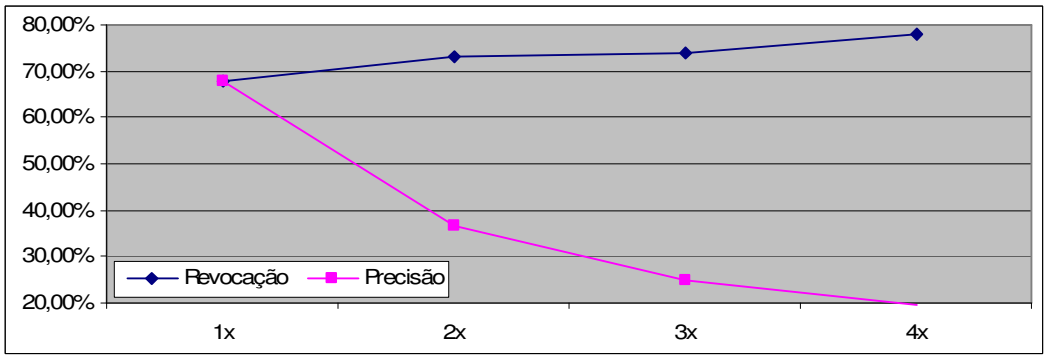


Gráfico 17 – Busca *wavelet* "por exemplo" de quadros sorteados

Fonte: elaborado pelo autor (2008).

No Gráfico 18, para o método baseado em *wavelet* “por rascunho”, embora os resultados não sejam expressivamente superiores que os demais, a revocação máxima é atingida logo no início, juntamente com a precisão máxima, o que é o caso ideal porque os quadros alvo esperados aparecem logo nas primeiras colocações. Em seguida, mesmo reduzindo-se a precisão, não há ganho na revocação.

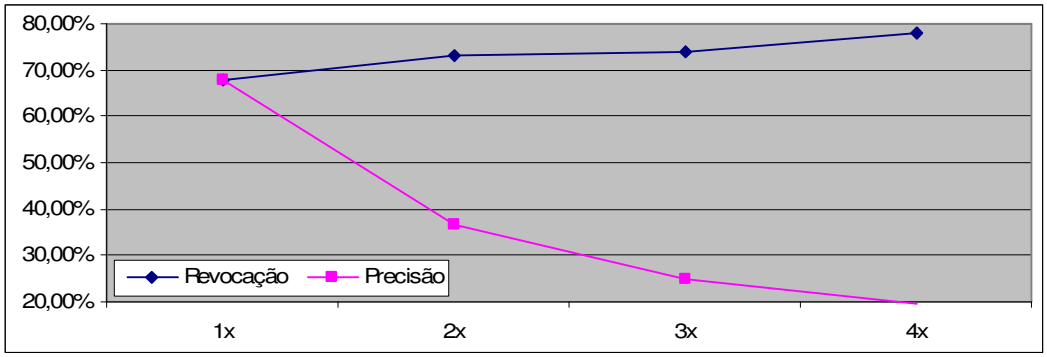


Gráfico 18 – Busca por rascunho modificado de quadros sorteados

Fonte: elaborado pelo autor (2008).

Houve quadros não encontrados nos três métodos, praticamente os mesmos exemplos, isso aconteceu por falha não no método de busca, mas sim na captura de alguns quadros-chave, indicando que esse é um ponto crítico do sistema, o que já fora discutido na relação de perda *versus* redundância na seção 5.1.1.4. Em compensação, os resultados das buscas apresentaram um desempenho muito bom, se for levado em consideração que a maior parte dos quadros não fora recuperada justamente por não ter sido indexados.

# 7 CONCLUSÕES

Os métodos e técnicas implementados e avaliados no ambiente apresentado, composto pelas ferramentas de Vídeo *Parsing* e Vídeo Oráculo, podem ser utilizados, em princípio, para processar e indexar qualquer categoria de vídeo digital. Em outras palavras, os resultados obtidos nesse trabalho não se restringem apenas à categoria de vídeo jornalístico (e, especificamente edições do Jornal Nacional), podendo ser potencialmente aplicados a outras categorias de vídeo, tais como documentários, filmes, seriados de TV e outros.

A redução do conjunto total de quadros do vídeo para um sumário compacto contendo um subconjunto desse total, ou seja, um sumário do vídeo, é importante para diminuir o custo computacional na extração de características que representam o conteúdo visual. Além disso, a representação compacta reduz o custo de armazenamento e o tamanho da base de dados para a indexação do conteúdo de um vídeo. Essas vantagens tornam o sistema de busca por similaridade visual mais eficiente e com poucas informações redundantes. Infelizmente, não é possível encontrar uma sumarização automática livre de perda de quadros-chave e ao mesmo tempo sem qualquer redundância de informação visual. O presente trabalho mostrou, porém, que sendo controlada, a relação perda *versus* redundância pode ficar dentro de um nível tolerável. Na pesquisa específica, esse nível de perda de quadros-chave atende às necessidades da aplicação de recuperação de conteúdo visual em vídeos digitais.

O uso de uma antena interna e de baixa qualidade no processo de captura das amostras de vídeo apresentou um pequeno ruído branco, o que não foi considerado grave (e foi até bem vindo, no sentido em que atesta a eficiência dos resultados de busca por similaridade), já que, com o ruído, mesmo trechos de vinhetas exatamente iguais (em termos de conteúdo gerado pelas emissoras de TV) apresentam diferenças visuais.

Os resultados das buscas visuais de quadros reforçam a tese de que a fase de captura de quadros-chave atende aos objetivos esperados. Quanto aos limiares propostos no presente trabalho para a melhor relação de perda *versus* redundância de quadros-chave, embora estes limiares não sejam considerados perfeitamente

válidos para todas as categorias de vídeo, pode-se dizer que a avaliação das características de cada método e a comparação entre os mesmos é uma contribuição importante, visto que os fundamentos e características apresentadas em cada método de sumarização nas diversas categorias de vídeo são consideradas universais.

Embora não tenha sido o objetivo inicial do presente trabalho as “buscas de quadros com tolerância a pequenas mudanças” carregam aspectos semânticos mesmo que em pequeno grau. Desse modo, o exemplo das buscas de quadros contendo o apresentador e a bandeira de um país que retornam quadros de notícias internacionais configuram uma busca que agregam semântica em algum grau. A avaliação ou quantificação do grau de semântica fica proposto como trabalho futuro.

Os métodos de busca de quadros mostraram-se satisfatórios tanto em termos de desempenho na qualidade das buscas, quanto em velocidade no retorno das respostas. Considerando-se a complexidade apresentada na indexação automática de conteúdo visual na mídia de vídeos e a recuperação desse tipo de conteúdo, os testes realizados num ambiente cem por cento implementado num PC (configuração média comercializada em 2007), com recuperação de quadros de vídeos visualmente similares em menos de um segundo, ou no máximo 20 segundos em casos extremos, isso a depender do método utilizado, é considerado um resultado bastante atraente visto que o conjunto indexado de vídeos nesse ambiente fora superior a 34 horas.

Os três métodos de busca: por características estatísticas; por assinatura *wavelet* “rascunho” e “por exemplo”, mostraram-se basicamente estáveis para a proposta de busca de quadros sorteados. Para que esses métodos tivessem melhores resultados seria necessário uma redução da taxa de perda de quadros do módulo de captura de quadros-chave realizado pelo vídeo *parsing*, o que por outro lado acarretaria mais redundância.

## 7.1 TRABALHOS FUTUROS

Como trabalhos futuros, projeta-se adquirir uma base de vídeos maior e com menos ruído. Mais testes podem ser feitos na avaliação do desempenho da captura de quadros-chave, bem como podem ser testadas outras categorias de vídeos, tais como filmes, seriados, documentários, vídeos educativos, vigilância dentre outros.

A base de dados gerada pelo vídeo *parsing* é uma fonte extremamente rica em conteúdo, podendo ser utilizada na mineração de dados e no aperfeiçoamento da indexação dos vídeos. Por exemplo, essa base pode ser explorada para segmentação temporal semântica do vídeo com a identificação de unidades lógicas tais como cenas, diálogos, comerciais, dentre outras entidades. A análise desses dados também pode permitir uma outra abordagem de indexação paralela à já existente, adicionando a técnica de estratificação. Com intervenção humana, os vídeos jornalísticos que apresentam repetição de conteúdo visual e padrões na programação podem ser automaticamente rotulados em *stratas*. Esse é o caso dos vídeos jornalísticos cujos padrões de cenas de vinhetas se repetem gerando quadros similares. Desse modo, o usuário poderia classificar um conjunto de quadros definindo sua semântica e a ferramenta de busca poderia gerar indexação baseada em estratificação marcando temporalmente as suas ocorrências na base de dados.

Os métodos de busca de quadros podem ser melhorados, dando suporte a busca de quadros similares com seleção manual feita pelo usuário de quais regiões espaciais da imagem fornecida como exemplo devem ser levadas em consideração e quais regiões devem ser descartadas pelo sistema de busca, desse modo, o usuário não fica obrigado realizar a busca com 100% do conteúdo visual da imagem de exemplo.

## REFERÊNCIAS


A DEVORE, Ronal; JAWERTH, Björn; LUCIER, Bradley. Image compression through wavelet transform coding. **IEEE Trans.:** Inform. Theory, Columbia, p.719-746, mar. 1992.

BIEDERMAN Irving. **Human image understanding: Recent research and a theory.** In: Papers from the second workshop Vol. 13 on Human and Machine Vision II. San Diego: Academic Press Professional, Inc. p. 13-57, 1986.

BORECZKY John S.; WILCOX Lynn D.. A Hidden Markov Model Framework for Video Segmentation Using Audio and Image Features. In: IEEE International Conference on Acoustics, Speech and Signal Processing, 1998. Washington: **IEEE Computer Society.** v. 6, p. 3741-3744, Mai. 1998.

BOVIK, Alan C.; GIBSON, Jerry D. (Ed.). **Handbook of Image and Video Processing.** Orlando: Academic Press, Inc., 2000.

CANTARELLI, E. M. P.; SOTT, J.. Indexação e Recuperação de Informações Digitais para Vídeo Usando MPEG-7. In: II Congresso Sul Catarinense de Computação, 2006, Criciuma. **Anais**… Criciuma: UNESC, 2006.

CARSON, Chad *et al*. Blobworld: image segmentation using expectation-maximization and its application to image querying. **IEEE Transactions, Pattern Analysis and Machine Intelligence**, v. 24, n. 8, p.1026-1038, ago. 2002.

CERNEKOVA Z.; NIKOU C.; PITAS, I.. Shot detection in video sequences using entropy based metrics. In: Proceedings. 2002 International Conference on Image Processing. 2002. Washington: **IEEE Computer Society**, v.3, p. 421-424, Jun. 2002.

CHRISTOPOULOS, C; SKODRAS, A; EBRAHIMI, T.. The JPEG2000 still image coding system: An overview. **IEEE Trans**. Consumer Electronics, v. 46, n. 4, p. 1103-1127, nov 2000.

CHUA Tat-Seng; CHEN Liping; WANG Jihua. Stratification Approach to Modeling Video. **Multimedia Tools and Applications.** Hingham: Kluwer Academic Publishers. v.6, e.1-2, p 79-97, Jan/Fev, 2002.

CHUI, Charles K.. **An Introduction to Wavelets, Volume 1.** San Diego: Academic Press Professional, Inc., 1992. 226 p., 1992.

DAUBECHIES, Ingrid. **Ten Lectures on Wavelets.** Philadelphia: Society For Industrial And Applied Mathematics, 1992.

DAVENPORT, Glorianna; SMITH, Thomas Aguirre; PINCEVER, Natalio. Cinematic Primitives for Multimedia. **IEEE Computer Graphics And Applications**, Los Alamitos: IEEE Computer Society Press, v. 11, n. 4, p.67-74, jul. 1991.

DESELAERS,T.. **Features for Image Retrieval.** Rheinisch-Westfalische Technische Hochschule, Technical Report, Aachen 2003.

DESELAERS, Thomas; KEYSERS, Daniel; NEY, Hermann. **Features for image retrieval: A quantitative comparison**. In: Congrès Pattern recognition. 2004, Tübingen: DAGM symposium, p. 228-236, 2004.

DIMITROVA, Nevenka *et al*. Applications of Video-Content Analysis and Retrieval. **IEEE Multimedia**, Los Alamitos, p.42-55, jul. 2002.

DREW M. S.; AU, J. **Video keyframe production by efficient clustering of compressed chromaticity signatures.** In: Proceedings of the eighth ACM international conference on Multimedia. New York, ACM. p. 365-367, Nov. 2000.

EIDENBERGER, Horst. **How Good are the visual MPEG-7 features?** In: Proceedings SPIE Visual Communications and Image Processing Conference. Lugano Switzerland: SPIE v. 5150, p. 476-488, jul. 2003.

EPSTEIN, Isaac. **Teoria da informação**. São Paulo: Ática, 1988.

FALOUTSOS, Christos Nick *et al*. Efficient and effective querying by image content. **Journal Of Intelligent Information Systems**, Hingham, p. 231-262. jul. 1994.

FENG, David; SIU, W. C.; ZHANG, Hong J., (Ed.). **Multimedia Information Retrieval and Management:** Technological Fundamentals and Applications. New York: Springer, 2003. (Signals and Communication Technology), 2003.

FLICKNER, Myron *et al*. **Query by image and video content: The QBIC system**. Mit Press, Cambrige, p.7-22, 1997.

FOOTE, J. T. **A Similarity Measure for Automatic Audio Classification.** In: Proceedings of the AAAI 1997 Spring Symposium on Intelligent Integration and Use of Text, Image, Video, and Audio Corpora. Stanford, 1997.

GERTZ, M. *et al.*. **Annotating Scientific Images: A Concept-based Approach.** In: PROCEEDINGS OF THE 14TH INTERNATIONAL CONFERENCE ON SCIENTIFIC AND STATISTICAL DATABASE MANAGEMENT. 2002, Washington: IEEE Computer Society, p 59-68, 2002.

GONZALEZ, R. C.; WOODS, R. E.. **Processamento de imagens digitais**. São Paulo: Edgard Blücher, 2000.

GUAN, Ling; KUNG, S. Y.; LARSEN, Jan (Ed.). **Multimedia Image and Video Processing: Video Modeling and Retrieval.** Boca Raton: Crc Press, Inc., 2000.

GUIMARÃES, Sílvio Jamil Ferzoli. **Video transition identification based on 2D image analysis.** 2003. 119 f. These (Phd) - Universidade Federal de Minas Gerais And Université de La Marne-la-vallée, Belo Horizonte, 2003.

GUPTA, Amarnath; JAIN Ramesh. Visual information retrieval. **Communications of the ACM.** New York: ACM, v. 4, e. 5, p. 70-79, Mai. 2007.

HANJALIC, A.; ZHANG, H. J. An integrated scheme for automated video abstraction based onunsupervised cluster-validity analysis. **IEEE Transactions:** Circuits and Systems for Video Technology, p.1280-1289, dez. 1999.

HANJALIC, A.; LAGENDIJK**,** R.L.**;** BIEMOND**,** J**.** Automated high-level movie segmentation for advanced video-retrieval systems. **IEEE Transactions:** Circuits and Systems For Video Technology, Vol. 9, n. 4, p. 580-588, Jun 1999.

IBM ALMADEN RESEARCH CENTER. **Almaden**. Disponível em: <http://www.almaden.ibm.com>. Acesso em: 09 mar. 2007.

IBM T. J. WATSON RESEARCH CENTER. **Marvel: Multimedia Analysis and Retrieval System**. Dispopnível em: <http://mp7.watson.ibm.com/marvel/> Acesso em: 18 jul. 2007.

JACOBS, Chuck; FINKELSTEIN, Adam; SALESIN, David. **Fast Multiresolution Image Querying.** In: Proc SIGGRAPH. 1995, New York: ACM, 1995. p. 277-286. Disponível em: <http://grail.cs.washington.edu/projects/query/>. Acesso em: 08 mar. 2007.

JEON, Jiwoon; LAVRENKO, Victor; MANMATHA, Raghavan, 2003, Toronto. **Automatic image annotation and retrieval using cross-media relevance models.** Toronto: Association For Computing Machinery, 2003.

SMITH, John R.; CHANG, Shih-Fu, 1997, Boston. **VisualSEEk: a fully automated content-based image query system.** New York: Acm, 1997.

KIM Hyount-Gook *et al*.. **Speaker Recognition Using MPEG-7 Descriptors.** In: Proceedings of EUROSPEECH. Germany: Department of Communication Systems Technical University of Berlin, 2003.

LEHANE B.; O'CONNOR N.; MURPHY N., **Dialogue scene detection in movies using low and mid-level visual features.** In: Proceedings of the International Conference Image and Video Retrieval and Mining. Quebec. p. 286-296. Out. 2004.

LI, Jia; WANG, James Z.. Automatic linguistic indexing of pictures by a statistical modeling approach. **IEEE Transactions On Pattern Analysis And Machine Intelligence**, Washington, v. 2003, n. 25, p.1075-1088, set. 2003.

LIENHART R.; KUHMÜNCH C.; EFFELSBERG W. **On the detection and recognition of television commercials.** In: Proceedings of the 1997 International Conference on Multimedia Computing and Systems. Washington: IEEE Computer Society p. 509-516, Jun. 1997.

LIMA P. C.. **Wavelets: Uma Introdução.** Matemática Universitária, 33, p. 13-44, 2002.

LIN Tong; ZHANG Hong-Jiang. **Automatic Video Scene Extraction by Shot Grouping.** in: Proc. 15th International Conference on Pattern Recognition, 2000. Washington: IEEE Computer Society v. 4, p. 39-42. mar. 2000.

LUI T., ROSENBERG C., ROWLEY H. A.. **Clustering Billions of Images with Large Scale Nearest Neighbor Search.** In: Proceedings of the Eighth IEEE Workshop on Applications of Computer Vision. Washington: IEEE Computer Society, p. 28, 2007.

MA, Wei-ying; MANJUNATH, Bangalore S. NeTra: a toolbox for navigating large image databases. **Special Issue On Video Content Based Retrieval**, New York, p.184-198, maio 1999.

MANZATO, M. G.; GOULARTE, R.. **Shot boundary detection based on intelligent systems.** In: XIII Simpósio Brasileiro de Sistemas Multimídia e Web, 2008, Vila Velha. Anais do XIV Simpósio Brasileiro de Sistemas Multimídia e Web, 2007, Gramado-RS. Proceedings of the XIII Brazilian Symposium on Multimedia and the Web, 2007.

MARTÍNEZ, JOSÉ M. **MPEG-7 Overview.** INTERNATIONAL ORGANISATION FOR STANDARDISATION. Disponível em: <http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm>. Acesso em: 12 dez. 2007.

MISITI, Michel *et al*. **Wavelet Toolbox User's Guide for use with Matlab.** Natick: The Mathwworks Inc., 2002.

MOORES, C.N. **Datacoding applied to mechanical organization of knowledge.** American Documentation 2, p. 20–32, 1951.

PFEIFFER, Silvia *et al*. **Abstracting Digital Movies Automatically.** Mannheim: University Of Mannheim, 1996.

PIMENTEL FILHO, C. A. F.; MONTALVÃO J.; REHEM NETO A.. **Um Estudo de Segmentação de Imagens Baseado em Textura.** In: III Simpósio Brasileiro de Sistemas de Informação, 2006, Curitiba. Anais do III Simósio Brasileiro de Sistemas de Informação, 2006. v. 01. p. 12-12.

PINQUIER Julien; SÉNAC Christine; ANDRÉ-OBRECHT Régine. **Speech and music classification in audio documents.** In: Acoustics, Speech, and Signal Processing, 2002. Proceedings ICASSP 2002).

RUI Yong; HUANG Thomas S.; MEHROTRA, Sharad. Constructing table-of-content for videos. **Multimedia Systems:** Special section on video libraries. Secaucus: Springer-Verlag New York, Inc. v. 7, e. 5, p. 359-368, Set. 1999.

RUI Yong; HUANG Thomas S.; MEHROTRA, Sharad. **Exploring video structure beyond the shots.** In: Proceedings of the IEEE International Conference on Multimedia Computing and Systems. Washington: IEEE Computer Society. p. 237-240, 1998.

SANTOS, C. A. S.; REHEM NETO, Almerindo Nascimento. **Uma Abordagem para Anotação em Vídeos Digitais com Aplicações em Telemedicina.** In: IV Workshop de Informática Médica, 2004, Brasília. IV Workshop de Informática Médica - WIM, 2004.

SANTOS, Tiago Teixeira. **Segmentação automática de tomadas em vídeo.** 2004. 77 f. Dissertação (Mestrado) - Universidade de São Paulo, São Paulo, 2004.

SBC - SOCIEDADE BRASILEIRA DE COMPUTAÇÃO (Org.). **Grandes Desafios da Pesquisa em Computação no Brasil**. Brasil, 2006. Disponível em: <www.sbc.org.br/index.php?language=1&content=downloads&id=272>. Acesso em: 12 dez. 2007.

SEINSTRA, Frank J. *et al*. High-Performance Distributed Video Content Analysis with Parallel-Horus. **IEEE Multimedia**, Los Alamitos, n. , p.64-75, out. 2007.

SIMÕES, Nielsen Cassiano. **Detecção de Algumas Transições Abruptas em Següencias de Imagens**. 2004. 52 f. Dissertação (Mestrado) - Unicamp, Campinas, 2004.

STOLLNITZ J. Eric; DEROSE D Tony; SALESIN H. David. Wavelets for computer graphics: A primer, Part 1. **IEEE Computer Graphics and Applications.** Los Alamitos: IEEE Computer Society Press, 1995. V. 15 n. 3, p. 76-84, mai 1995.

SUNDARAM H.; CHANG Shih-Fu. **Video Scene Segmentation Using Video and Audio Features.** In: IEEE International Conference on Multimedia and Expo. New York: Elsevier Science Inc. v. 2, p. 1145-1148. Jul. 2000.

SWAIN, Michael J.; BALLARD, Dana H.. Color indexing. **International Journal Of Computer Vision**, p. 11-32. nov. 1991.

TAHAGHOGHI S. M. M. *et al*.. **Video cut detection using frame windows.** In: Proceedings of the Twenty-Eighth Australasian Computer Science Conference (ACSC 2005), volume 38, Newcastle: Australian Computer Society Inc., v.38, p. 193-199, 2005.

TIAN, Qi. A Foundational Perspective for Visual Information Retrieval. **IEEE Multimedia**, Los Alamitos, v. 2, n. 13, p.90-92, abr. 2006.

TRUONG, Ba Tu. **In Search of Structural and Expressive Elements in Film Based on Visual Grammar.** 2004. 197 f. These (Phd) - Curtin University, Sydney, 2004.

TSINARAKI Chrisa; CHRISTODOULAKIS Stavros. **Semantic User Preference Description in MPEG-7/21.** In: Proc. of HDMS 2005, Athens. Ago 2005.

WANG, Yao; LIU, Zhu; HUANG, Jin-cheng. Multimedia content analysis-using both audio and visual clues. **IEEE Signal Processing Magazine**, p.12-36, dez. 2000.

WEN X., HUFFIMIRE T. D., FINKELSTEIN A. **Wavelet-Based Video Indexing and Querying for a Smart VCR.** Princeton University, 1998.

ZHANG H. J. *et al*.. **Automatic parsing of news video.** In: Proceedings of the International Conference on Multimedia Computing and Systems. Washington: IEEE Computer Society, . p. 45-54. Mai. 1994.

ZHANG H. J.; WANG J. Y. A.; ALTUNBASAK Yucel. **Content-Based Video Retrieval and Compression: A Unified Solution.** In: Proceedings of the 1997 International Conference on Image Processing (ICIP '97). Washington: IEEE

Computer Society, p. 13, 1997.

ZIVIANI, Nivio *et al*.. **Projeto de Algoritmos com implementações em Pascal e C**. São Paulo: Pioneira, 2004.