1
Introdução

1.1
Motivação

A partir de 1999, com a aprovação do padrão de compressão de imagens MPEG-4 da ISO/IEC (*International Organization for Standardization/International Electrotechnical Commission*), a codificação orientada por objeto proporcionou, além da melhoria da qualidade subjetiva da imagem, facilidades decorrentes da manipulação independente dos objetos dentro de uma mesma cena. Essas facilidades visaram principalmente à escalabilidade e à interatividade nos serviços avançados de multimídia, ao armazenamento e recuperação de dados baseados em objetos, à representação e manipulação dos objetos em uma cena em pós-produções de TV e cinema, aos jogos de computador que exploram a mistura de objetos naturais e sintéticos e às aplicações multimídia móveis. O padrão MPEG-4 é constituído de várias 'partes' relacionadas, sendo a codificação visual orientada por objeto realizada pela parte 2. A parte 2 do MPEG-4 inclui os processos de segmentação e codificação da forma dos objetos, estimação e compensação de movimento e codificação de intensidade (textura) intra e interquadros.

O padrão H.264, desenvolvido a partir de 1998 com a junção do grupo VCEG - *Video Coding Experts Group* - da ITU (*International Telecommunication Union*) ao grupo MPEG - *Moving Pictures Expert Group* - da ISO/IEC, trata-se, na verdade, de uma das partes adicionadas mais tarde ao padrão MPEG-4. O MPEG-4 parte 10, como também é conhecido, proporciona economias de taxas de bits da ordem de 38%, se comparado ao MPEG-4 parte 2. Contudo, não apresenta as mesmas facilidades relacionadas à codificação orientada por objeto.

A motivação para esta tese é a melhoria dos ganhos de codificação de textura intra-quadro da imagem, que é um dos tópicos tratados pelos codificadores orientados por objeto. Busca-se a recuperação de objetos de

forma arbitrária com uma melhor qualidade subjetiva, ao mesmo tempo em que se possibilita tratar os objetos de forma independente, mantendo as facilidades de escalabilidade e interatividade com o conteúdo visual, apresentadas pelo padrão MPEG-4 parte 2.

Na Seção 1.2, é apresentado um histórico da evolução dos principais padrões de codificação de imagem, desde o JPEG, dedicado exclusivamente aos quadros de imagem estática, ao MPEG-4, atualmente cogitado para a utilização em TV digital. Também é realizada uma descrição sucinta comparativa entre as partes 2 e 10 do MPEG-4, que são às partes relacionadas à codificação visual dos objetos. Finalmente, na Seção 1.3 são listados os objetivos desta tese, cuja organização está descrita na Seção 1.4.

1.2 A Evolução dos Padrões de Codificação de Imagem

Nos últimos quinze anos, tem sido notável o crescente interesse na tecnologia de codificação digital de imagens estáticas e em movimento e suas aplicações em comunicações visuais. As taxas de transmissão a que estão limitadas as Redes de Telefonia Comutada (RTC), as Redes Digitais de Serviço Integrado (RDSI), as redes ATM, as redes móveis, os canais digitais de satélite, as atuais redes de banda larga com fio como xDSL (*Digital Subscriber Loop*), FTFC (*Fiber to Fiber Curb*), HFC (*Hybrid Fiber Cable*) e FTFH (*Fiber to Fiber Home*) [1], sem contar as redes de acesso fixas sem fio, especialmente as de banda larga, B-FWANs [1], apresentam um desafio às comunicações de vídeo digital, tornando cada vez mais importante a codificação de vídeo a baixas taxas. Dentre as aplicações que requerem a redução da largura da banda do canal de transmissão das imagens, podemos citar: SDTV - *Standard Definition TV*, HDTV - *High Definition TV*, vídeoconferência, monitoração de segurança, jogos interativos, teleshopping, sensoriamento remoto via satélite, radar, sonar e controle de veículos aéreos não tripulados. Também existem as aplicações de compressão de imagens que têm como principal objetivo reduzir a quantidade de memória para armazenamento. Dentre elas, podemos citar as imagens com finalidade médica, como a tomografia computadorizada, a ressonância magnética e a radiologia digital, além de jogos de entretenimento em CD-ROM, mapas meteorológicos e geológicos, documentação educacional, científica, comercial, artística e histórica.

Em virtude da necessidade de padronização de métodos de codificação de imagem, surgiram grupos de trabalho nos principais organismos de

normalização, como a ISO, a IEC e a ITU, com os órgãos ITU-T, dedicado à elaboração de normas na área de Telecomunicações, e ITU-R, voltado para a área de radiodifusão. Esses grupos desenvolveram vários padrões internacionais para a compressão de imagem e vídeo, visando a diferentes tipos de aplicação.

Dentre os principais padrões, destacam-se o JPEG - *Joint Photograph Experts Group* - (em 1991) [2]-[4] e o JPEG 2000 (em 1997) [5],[6], ambos da ISO/IEC, para imagens estáticas; o H.261 (em 1990) e o H.263 (em 1995) [3]-[12], ambos da ITU-T; e os padrões MPEG - *Motion Picture Experts Group* - 1 (em 1992), 2 (em 1994) e 4 (em 1999), da (ISO/IEC) [3],[12]-[31]. Em 1998, o VCEG/ITU-T iniciou o projeto do H.26L e em dezembro de 2001, o VCEG e o MPEG se uniram para formar o JTV - *Joint Video Team* - com a tarefa de finalizar o novo padrão de codificação de vídeo H.264/AVC [32]-[38]. Esse padrão também é conhecido por MPEG-4 parte 10 ou H.26L, o que tem gerado algumas confusões de nomenclatura. Todos esses padrões, à exceção do JPEG 2000, exploram redundâncias espaciais através de codificação intraquadro usando DCT (*Discrete Cosine Transform*). Os padrões para codificação de vídeo exploram ainda as redundâncias temporais através de codificação preditiva interquadros com compensação de movimento, que estima o deslocamento de um quadro para o outro e transmite o vetor de movimento como informação paralela, em adição à imagem de erro de predição [4],[8],[12],[27],[28]. Essa imagem de erro é codificada por um codificador intraquadro que também explora as redundâncias espaciais.

A padronização dos algoritmos de compressão de vídeo foi iniciada pelo grupo CCITT- *International Committee on Telegraph and Telephones* - a partir de 1980, visando a aplicações de teleconferência, a taxas de 2,048 e 1,544 Mbits/s, para os sistemas de TV com 625 linhas/50Hz e 525 linhas/60Hz, respectivamente. Com a definição das RDSI, houve uma motivação para a padronização de técnicas de compressão a taxas de $p\times64$ kbits/s, onde p assume o valor de 1 a 30 (número de canais RDSI). Esse trabalho culminou com a produção da recomendação H.261, em 1990 [7]-[9]. O padrão H.261 pode utilizar dois formatos de vídeo: QCIF (*Quarter CIF* - 176×144 pixels de luminância) e CIF (*Common Intermediate Format* - 352×288 pixels de luminância), que é opcional. A informação de cor é enviada com metade da resolução empregada para a luminância. Esse padrão usa DCT, DPCM (*Differential Pulse Code Modulation*) temporal e compensação de movimento usando vetor com resolução de 1 pixel.

As atividades de padronização da ISO iniciaram-se em 1982, focando,

inicialmente, a codificação de imagens com textura suave (em tons de cinza ou colorida). Em 1986, membros da ISO e do CCITT uniram-se para formar o *Joint Photograph Experts Group* - JPEG. O compressor JPEG é uma ferramenta de compressão de imagens estáticas para propósitos gerais [2]-[4]. Permite tanto a compressão sem perdas, onde os pixels são codificados através de um esquema preditivo (usando DPCM), quanto a compressão com perdas (usando DCT). Em seu modo de compressão sem perdas, permite taxas de compressão da ordem de 2:1. No modo de compressão com perdas, as taxas de compressão variam entre 5 e 20. Consiste basicamente de codificação DCT, seguida de quantização escalar e codificação de Huffman ou aritmética dos coeficientes, após varredura em zig-zag. O padrão JPEG 2000, concluído em 1997, não é compatível com o JPEG convencional, uma vez que utiliza *wavelet* ao invés da DCT [5],[6].

Em 1992 foi aprovado o padrão MPEG-1, desenvolvido para compressão de vídeo a taxas entre 1 e 1,5 Mbits/s, tendo como principais alvos o armazenamento em CD-ROM com qualidade comparável ao VHS [3],[4],[13] e a transmissão em canais de comunicação de banda estreita como as Redes Digitais de Serviço Integrado (RDSI). O grande desafio dos algoritmos MPEG foi atender a especificação de qualidade (o que demanda uma compressão alta, não obtida usando-se apenas codificação intraquadro), ao mesmo tempo em que provê acesso randômico (o que é mais facilmente obtido com codificação intraquadro pura). Isso requer um bom equilíbrio entre codificação intraquadro e interquadros. Na codificação interquadros, a compensação de movimento pode ser feita a partir de um quadro passado (modo preditivo) ou de quadros passados e futuros (modo bidirecional). O MPEG-1 também foi o primeiro padrão a integrar a codificação de áudio e vídeo, extraindo e demultiplexando essas duas informações para que fossem tratadas por decodificadores apropriados [13]. Essas e outras características tornaram o padrão MPEG-1 o formato para a distribuição de material de vídeo pela Web, representando a primeira oportunidade concreta encontrada pela indústria de microeletrônica para investir em vídeo digital.

O padrão MPEG-2 [3],[4],[13] foi aprovado em 1994, tendo sido desenvolvido para vídeo cm vários níveis de qualidade e aplicações em radiodifusão terrestre ou por satélite, TV a cabo, estúdio e HDTV, a taxas entre 1,5 Mbits/s e até aproximadamente 35 Mbits/s. Além disso, incluiu elementos necessários para a televisão interativa. Os princípios básicos são os mesmos dos padrões anteriores, mas o MPEG-2 é capaz de codificar seqüências de imagens representadas por campos (as linhas pares de um quadro de vídeo formam um campo e as linhas ímpares constituem outro

campo).

Os terminais que utilizam padrões MPEG (como televisores, computadores e telefones celulares) podem variar significativamente devido às diferenças de preço e aplicação, apresentando capacidades de processamento distintas. Por esse motivo, cada terminal pode implementar um ou mais subconjuntos do padrão completo, denominados 'profiles'. O 'profile' SSP - *Spatial Scalable Profile* - é suportado pelo MPEG-2, para o qual se transmitem uma camada-base e uma camada de detalhes. O SSP gera imagens de baixa resolução a partir de filtragem passa-baixas e sub-amostragem do sinal de vídeo original. As imagens de baixa resolução são codificadas separadamente e o código de bits resultante representa a camada-base do MPEG-2 SSP. Ainda no codificador, a camada-base é decodificada e as imagens de baixa resolução reconstruídas são interpoladas, sendo também usadas para gerar um sinal de predição da imagem de alta resolução. O erro de predição resultante é codificado na camada de detalhes a uma taxa de amostragem igual à do sinal de vídeo original. Portanto, o número total de amostras codificadas excede o número de amostras do vídeo original em número igual à quantidade de amostras codificadas na camada-base, o que provoca uma sobre-taxa de 66 a 70 % em relação ao MPEG-2 não escalável [29].

Objetivando a redução dessa sobre-taxa, em [29] propôs-se a escalabilidade multi-resolução, onde são transmitidas duas seqüências de bits, resultantes da decomposição da imagem em quatro subbandas de freqüências espaciais. A subbanda de baixas freqüências representa a camada-base e as demais representam a camada de detalhes, que fornece dados adicionais necessários à reprodução da imagem com a resolução completa. A subbanda de baixas freqüências e as três subbandas de alta freqüência (amostradas com 1/4 da resolução) são codificadas usando técnicas baseadas em DCT, compatíveis com o MPEG-2 não escalável. A camada-base pode ser decodificada independentemente da camada de detalhes, de forma que terminais de baixa resolução possam mostrar imagens de baixa resolução. Também foi introduzida a escalabilidade SNR - *Signal-to-Noise Rate*, onde os coeficientes DCT da camada-base são quantizados com menor precisão, e a escalabilidade temporal, onde são previstas diferentes taxas de transmissão dos quadros de imagem. A transmissão escalável é útil em ambientes suscetíveis a erros, porque a camada-base pode ser bem protegida contra erros e perdas de transmissão. A sobre-taxa introduzida pela escalabilidade multi-resolução foi de apenas 5 a 9 %, em relação ao MPEG-2 não escalável. A Tabela 1.1 fornece dados experimentais de codificação (sob requisitos de qualidade constante) da seqüência de teste ITU-R *FLOWER GARDEN*

usando o MPEG-2 não escalável, o MPEG-2 SSP e o MPEG-2 usando a escalabilidade multi-resolução - EMR - [29].

Tabela 1.1: Resultados experimentais para codificação com qualidade constante usando a seqüência *FLOWER GARDEN* [29]

RPR	**30 dB**		**32 dB**	
	Mbits/s	**(%)**	**Mbits/s**	**(%)**
MPEG-2 não escalável	4,14	100	6,26	100
MPEG-2 SSP	7,06	170,1	10,43	166,6
MPEG-2 com EMR	4,44	107,4	6,61	105,5

O MPEG-2 foi adotado como padrão de TV digital pelo projeto DVB (*Digital Video Broadcasting*), do qual fazem parte, em mais de 30 países, cerca de 220 organizações que incluem transmissoras de TV, operadoras de rede, fabricantes e órgãos reguladores [14]. O MPEG-2 também é o padrão recomendado para DVD (*Digital Video Disc*). Contudo, embora ele tenha sido a base inicial dos sistemas de televisão digital via satélite (DVB-S), cabo (DVB-C) e terrestre (DVB-T), outros meios de transmissão - como xDSL ou UMTS - requerem taxas de bits muito menores.

Aprovado em 1995, o H.263 [3],[4],[10],[11] é um codificador de vídeo baseado nas mesmas técnicas dos padrões anteriores, porém visando às aplicações a taxas abaixo de 64 kbits/s, especialmente vídeo-telefonia. Na verdade, o H.263 foi otimizado para taxas abaixo de 28,8 kbits/s, possibilitando a transmissão de informação áudio-visual em canais de banda estreita, como 9,6 kbits/s [4],[10]. Para essas taxas, os algoritmos até então existentes, H.261, MPEG-1 e MPEG-2, produziam efeitos indesejáveis de blocagem ou então requeriam operações a baixas taxas de quadros, resultando em baixa resolução temporal [10]. Entretanto, algumas melhorias em relação às estratégias de compensação de movimento do H.261 (como o uso de meio pixel de resolução para a compensação de movimento, que também pode ser feita no modo preditivo ou no modo bidirecional) [4],[15],[26] e também a preocupação com a codificação sem perda dos símbolos a serem transmitidos (através da utilização de um codificador aritmético), permitiram uma qualidade de imagem melhor a taxas mais baixas (comparado ao H.261, o H.263 requer cerca de metade da taxa de bits para a mesma qualidade) [12]. O codificador pode operar sobre cinco diferentes formatos de vídeo: além dos formatos QCIF e CIF (opcional) previstos no H.261, incorpora também os formatos SQCIF, com aproximadamente a metade da resolução do QCIF (128×96 pixels) e os formatos 4CIF (704×576 pixels) e 16CIF (1408×1152 pixels), todos opcionais, com respectivamente 4 e 16 ve-

zes a precisão do CIF. Embora ele tenha sido otimizado para taxas abaixo de 28,8 kbits/s, mais tarde foi verificado que ele supera o desempenho do H.261 também a taxas muito mais altas, tendo sido realizadas comparações até 600 kbits/s [10]. Isso tornou esse padrão competitivo com outros padrões que trabalham a taxas mais altas, como os padrões MPEG [10]. Assim, no estágio final de uso do H.263, a sua utilização a taxas usadas em RDSI também foi considerada.

O MPEG-4 é um projeto da ISO/IEC [4],[16]-[24] que foi desenvolvido a partir de 1994, de forma a equilibrar o ganho de codificação, a complexidade e o custo de implementação, objetivando beneficiar-se da tecnologia para o projeto de circuitos integrados. O padrão foi concebido para ser a linguagem universal para aplicações de TV e cinema (a taxas de até 2 Mbits/s), telefonia celular (a taxas entre 5 e 64 kbits/s) e Internet (a taxas baixas, como 28,8 kbits/s), tendo se tornado uma recomendação internacional em 1999.

O MPEG-4 consiste de partes relacionadas, mas que podem ser implementadas individualmente ou combinadas com outras partes. A base do padrão é formada pelo Sistema (parte 1), Visual (parte 2), Áudio (parte 3) e DMIF - *Delivery Multimedia Integration Framework* (parte 6), que define a interface entre a aplicação e as redes de distribuição. A parte 4 define como testar uma implementação MPEG-4, a parte 5 contém o *software* de referência e a parte 7, uma proposta de implementação do codificador, porém não otimizada em termos de complexidade e qualidade.

O MPEG-4 parte 2 aplica o conceito de codificação de vídeo orientada por objeto, onde os sinais de vídeo são compostos de objetos distintos, descritos por sua informação de forma, movimento e textura [4]. O padrão compõe a cena a partir de objetos, aos quais podem ser associados dados, direitos de propriedade intelectual e indicadores de qualidade de serviço. Permite-se a codificação simultânea de imagens naturais e sintéticas, com envio de parâmetros para a calibração e animação dos personagens sintéticos. É possível fazer o *download* dos modelos de animação ou usar um modelo *default*.

Portanto, o MPEG-4 parte 2 deve ser capaz de codificar, decodificar e reconstruir objetos independentes em uma mesma cena, proporcionando características como escalabilidade e interatividade. Incorpora uma ferramenta de edição que faz um acesso aleatório no *bitstream* para permitir *fast-forward* e *fast-reverse*. A codificação de forma está prevista como auxiliar na descrição dos objetos e um mapa binário define se um pixel pertence ou não ao objeto.

O 'profile' SSP - *Spatial Scalable Profile*, introduzido no MPEG-2, também é suportado pelo MPEG-4 parte 2 [22]. Prevendo escalabilidade de taxas de transmissão, o MPEG-4 parte 2 possui um descritor de qualidade genérico (QoS - Qualidade de Serviço), visando à sua utilização em redes. Melhoramentos posteriores procuraram diminuir a sobre-taxa introduzida pelo 'profile' SSP, reduzindo a taxa de bits necessária à transmissão da camada-base, que demanda cerca de 60 % do total de bits. Com esse objetivo, foi introduzida, além da escalabilidade SNR - *Signal to Noise Rate* - [29], a escalabilidade espaço-temporal [30], reduzindo a sobre-taxa a valores menores que 6 % e 15 %, respectivamente. A escalabilidade SNR no padrão MPEG-4 amostra a sub-banda de baixas freqüências com metade da resolução e transmite os dados adicionais para refinamento da camada-base na camada de detalhes. Contudo, essa solução perde a compatibilidade completa com o padrão MPEG-2 [29]. Na escalabilidade espaço-temporal, por sua vez, a compatibilidade é mantida e a camada-base corresponde à seqüência de bits resultantes da redução das resoluções espacial e temporal da imagem [30]. Nesse caso, a seqüência de imagens é analisada por um banco de filtros 3D separável (temporal, vertical e horizontal), gerando duas sub-bandas temporais L_T (*low-frequency-temporal*) e H_T (*high-frequency-temporal*), cada uma delas com quatro subbandas espaciais (LL,LH,HL e HH). A camada-base é composta pela subbanda espacial LL da subbanda temporal L_T e a camada de detalhes é composta pelas subbandas espaciais LH, HL e HH da subbanda temporal L_T e pela subbanda espacial LL da subbanda temporal H_T. As demais subbandas espaciais da subbanda H_T são desprezadas.

O MPEG-4 parte 2 permite que codificadores de diferentes complexidades gerem um *bitstream* válido e significativo [30]. Da mesma forma, permite que os decodificadores menos complexos processem apenas uma parte do *bitstream*. Na escalabilidade espacial, estão previstos onze níveis de resolução para a textura em imagens paradas e três níveis para seqüências de vídeo e a escalabilidade de qualidade permite a partição do *bitstream* em camadas com taxas de bits diferentes. O padrão ainda conta com a escalabilidade baseada em objeto, permitindo estender os tipos de escalabilidade descritos ao tratamento de objetos específicos.

A primeira aplicação a adotar o MPEG-4 parte 2 foi a Internet móvel. Os fabricantes de celulares reagiram rapidamente e introduziram no mercado terminais baseados na tecnologia MPEG-4. Posteriormente, a Apple, Cisco, IBM, Kasena, Philips e Sun fundaram o *'Internet Streaming Media Alliance'* com o objetivo de definir um padrão aberto para redes banda-larga

baseado em MPEG-4. Também a indústria de eletrônica de consumo, como a Sony, já está fabricando equipamentos de entretenimento para decodificar MPEG-1, MPEG-2 e MPEG-4.

Recentemente, foram adicionadas as seguintes partes ao padrão MPEG-4: parte 8, que define o mapeamento do *bitstream* MPEG-4 em redes IP; a parte 9, que descreve a referência para o *hardware* MPEG-4; a parte 10, AVC - *Advanced Video Coding*; a parte 11, que faz a descrição das cenas; a parte 12, que é o formato de mídia da ISO; a parte 13, que é uma extensão IPMP - *Intellectual Property Management and Protection*; a parte 14, que é o formato MP4; a parte 15, que é o formato AVC; e finalmente, a parte 16, que é uma extensão dos aplicativos de animação e multi-usuários [43].

As facilidades previstas no MPEG-4 parte 2, que visaram principalmente aos serviços avançados de multimídia, como a interatividade com o conteúdo visual e as bibliotecas digitais, ao armazenamento e recuperação de dados baseados em objetos, à representação e manipulação dos objetos em uma cena em pós-produções de TV e cinema, aos jogos de computador que explorem a mistura de objetos naturais e sintéticos e às aplicações multimídia móveis, não foram incorporadas pelo MPEG-4 parte 10.

O padrão MPEG-4 parte 10 (ou H.264/AVC), cujo desenvolvimento iniciou em 1998 a partir da junção do VCEG-ITU-T - *Video Coding Experts Group* - com o MPEG-ISO/IEC - *Moving Pictures Expert Group*, concentrou-se em otimizar os algoritmos de compressão, buscando melhorias na qualidade subjetiva do vídeo codificado a taxas baixas e fazendo uso dos grandes avanços no desenvolvimento de processadores dedicados rápidos a partir da padronização do MPEG-2 [13],[34].

O projeto do H.264/AVC ou MPEG-4 parte 10 envolve a VCL - *Video Coding Layer*, que representa o conteúdo de vídeo de forma eficiente, e a NAL - *Network Abstraction Layer*, que formata a representação do vídeo em VCL e acrescenta informações no cabeçalho que permitam a utilização em meios de transmissão ou armazenamento específicos [32]-[40]. A versão final do *software* de referência foi disponibilizada em julho de 2003.

O VCL, assim como qualquer padrão anterior, desde o H.261, utiliza a abordagem híbrida da codificação de vídeo baseada em blocos, ou seja, combina a predição interquadros, que explora as redundâncias temporais, com a codificação por transformada do erro de predição residual, para explorar as redundâncias espaciais. Em seguida, aplica-se uma codificação de entropia adaptativa sofisticada, baseada em conteúdo. Não há um elemento único no VCL que seja o responsável pela melhoria drástica da eficiência de codificação, comparando-o aos outros padrões. Na verdade, existe uma plu-

ralidade de pequenos melhoramentos que juntos, proporcionam o aumento do ganho. A Tabela 1.2 fornece os valores médios de economia de taxas de bits proporcionada pelos padrões H.264/AVC (ou MPEG-4 parte 10), MPEG-4 parte 2 e H.263 em relação aos padrões anteriores [32].

Tabela 1.2: Economia média de taxas de bits em comparação a padrões anteriores [32]

PADRÃO	MPEG-4 (parte 2)	H.263	MPEG-2
H.264/AVC ou MPEG-4 (parte 10)	38,62 %	48,80 %	64,46 %
MPEG-4 (parte 2)	-	16,65 %	42,95 %
H.263	-	-	30,61 %

Algumas comparações entre as principais ferramentas empregadas pelos codificadores do MPEG-4 parte 2 e MPEG-4 parte 10 (ou H.264/AVC) são descritas nos parágrafos seguintes.

Com relação à codificação de textura no modo INTRA, onde nenhuma informação fora do quadro é utilizada, o MPEG-4 parte 2 prevê a utilização de *wavelets*, permitindo também a DCT baseada em blocos 8×8 e a SA-DCT. Nenhuma filtragem 'deblocking' é feita. No MPEG-4 parte 10, os blocos são 4×4 (áreas de alta textura) ou 16×16 (áreas de baixa textura) [36]. Um bloco é estimado usando-se blocos espacialmente vizinhos anteriormente codificados [32]. O codificador seleciona quais e como esses blocos vizinhos são utilizados na predição INTRA. Esse processo é realizado também no decodificador, a partir da informação paralela transmitida. O erro de predição - que é a diferença entre o quadro original e o quadro estimado - é transformado usando-se DCT inteira. Se o bloco de suporte é 4×4, o bloco 2×2 formado pela junção de quatro componentes DC da informação de croma são novamente transformados. No caso de um bloco de 16×16, o bloco 4×4 formado por coeficientes DC provenientes da transformada aplicada à luminância também são novamente transformados. Esse padrão conta com um filtro 'deblocking' adaptativo, cuja intensidade pode ser definida por parâmetros específicos [38].

Para a compensação de movimento, o MPEG-4 parte 2 prevê macroblocos de 16×16, com quatro vetores de compensação de movimento para cada macrobloco (um vetor para cada bloco 8×8), ao passo que no MPEG-4 parte 10 os macroblocos podem ser de 16×16, 16×8, 8×16 ou 8×8 pixels [37]. No caso de um macrobloco 8×8, pode-se criar um novo elemento para a definição de sub-macroblocos com 8×8, 8×4, 4×8 ou 4×4 pixels, para a compensação de movimento. Então, quando o macrobloco 8×8 é utilizado com as partições 4×4, até 16 vetores de movimento

podem ser transmitidos para cada conjunto de 16×16 pixels. A Figura 1.1 detalha a segmentação dos macroblocos para a compensação de movimento no MPEG-4 parte 10. A escalabilidade no MPEG-4 parte 10 é definida a partir da escolha dos macroblocos utilizados, sendo mais eficiente que a escalabilidade do MPEG-2 [40].

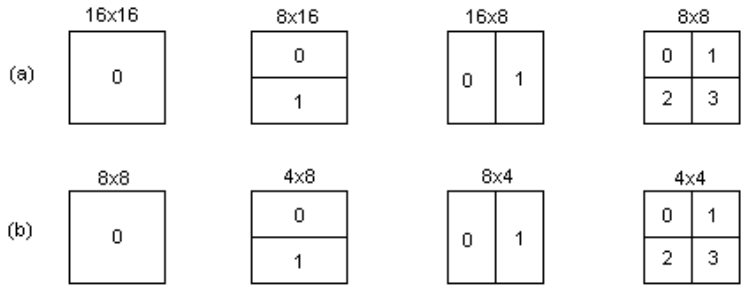

Figura 1.1: Segmentação dos macroblocos para a compensação de movimento no MPEG-4 parte 10: (a) segmentação dos macroblocos; (b) segmentação das partições 8×8

O MPEG-4 parte 2 não prevê a organização dos macroblocos, enquanto o MPEG-4 parte 10 organiza-os em fatias que podem ser decodificadas de forma independente, cuja ordem de transmissão é definida por um mapa de macroblocos. Em especial, especifica-se a ferramenta FMO - *Flexible Macroblock Ordering*, que faz a interpolação das informações dos macroblocos vizinhos para estimar um outro cuja fatia foi perdida durante a transmissão, o que aumenta a robustez da compressão a erros de transmissão. No MPEG-4 parte 2, os macroblocos podem ser do tipo I (intra), P (modo de predição) ou B (modo de interpolação). O MPEG-4 parte 10 prevê ainda outros dois tipos: SP (*switching P*) e SI (*switching I*), definidos de forma a permitir a comutação eficiente entre *bitstreams* codificados em várias taxas de bits.

Na codificação dos *P-frames* no MPEG-4 parte 10, mais de um quadro anterior podem ser usados como referência para a compensação de movimento. O índice do quadro de referência é transmitido para cada bloco de luminância, mas no caso da utilização de sub-macroblocos, todos devem ter o mesmo índice. Tanto o MPEG-4 parte 2 quanto o MPEG-4 parte 10 usam uma precisão de 1/4 de pixel para os vetores de movimento. Na compensação de movimento tipo B, o MPEG-4 parte 2 utiliza dois vetores de compensação distintos, mas com peso igual, e o MPEG-4 parte 10 faz uma

ponderação entre eles [35]. O *buffer* do MPEG-4 parte 2 possui os mesmos requisitos do MPEG-2, apresentando baixo atraso. Já no MPEG-4 parte 10, há a necessidade de gerenciamento de *buffer* para tratar a predição usando múltiplos quadros.

Enquanto o MPEG-4 parte 2 aplica ferramentas específicas para o tratamento do vídeo entrelaçado ou progressivo, o MPEG-4 parte 10 permite mesclar os dois tipos. Ambos os padrões utilizam quantização escalar com varredura em zig-zag, podendo-se usar um padrão de varredura alternativo no caso de codificação em campos de imagens entrelaçadas [36].

Para a codificação de entropia, o MPEG-4 parte 2 emprega o código de corrida de zeros e o código de Huffmann, ao passo que esse processo, no MPEG-4 parte 10, é muito mais sofisticado. Na codificação dos coeficientes quantizados, é usado um código de tamanho variável e adaptativo de acordo com o contexto (CAVLC - *Context-Adaptive Variable Length Code*). Uma alternativa, mais eficiente porém mais complexa, é empregar a codificação aritmética baseada em contexto [35].

Atualmente, tanto o MPEG-4 parte 2 quanto o MPEG-4 parte 10 estão sendo considerados para aplicações de TV digital, mas a sua inclusão nos receptores ainda tem um custo alto para o consumidor, mesmo com a disponibilidade de circuitos integrados aplicados. Espera-se que o padrão de codificação de vídeo H.264/AVC (ou MPEG-4 parte 10), com a sua significativa economia de banda de freqüência (vide Tabela 1.2), substitua o padrão MPEG-2 nos sistemas de vídeo digital. Os ganhos em taxas de bits, mesmo que com um aumento da complexidade dos algoritmos de codificação e decodificação, encorajam as transmissoras de TV a começar a usar o novo padrão, aproveitando a banda de freqüência disponível para a ampliação de outros canais ou para serviços interativos [40]. Com os ganhos de codificação do H.264/AVC, filmes em HDTV com resolução máxima hoje podem ser armazenados em DVDs. E ainda, a possibilidade de que as transmissoras de TV e a Internet utilizem o mesmo formato de codificação de vídeo irão criar novas possibilidades de serviço. A migração completa para esse novo algoritmo de codificação poderá levar alguns anos, dada a larga utilização do MPEG-2 hoje, no mercado. Por esse motivo, há uma carência de tecnologia que transcodifique o MPEG-2 para o H.264/AVC e *vice-versa*.

Até o momento, a indústria adotou apenas uma pequena porção do padrão completo especificado pelo *Motion Picture Experts Group*, tendo sido o MPEG-4 parte 2 associado apenas à compressão a taxas baixas, com complexidade relativamente baixa [31]. As maiores inovações, conceitos e tecnologias previstos no MPEG-4 parte 2, como a descrição de cenas e

objetos em um ambiente multimídia e interativo, ainda não encontraram aplicação no mercado.

Em conseqüência da ampliação do potencial de aplicação do MPEG-4 parte 2, que na realidade é um grande conjunto de ferramentas para representar conteúdo multimídia comprimido, escalável, com múltiplas aplicações e focado na redução de custos, a codificação de imagem orientada por objeto tem sido objeto de grande esforço de pesquisa. O estudo desses codificadores inclui os temas: segmentação e codificação da forma dos objetos segmentados, estimação e compensação de movimento e codificação de intensidade (textura) intra e inter-quadros.

1.3 Objetivos

Esta tese tem por finalidades pesquisar os fundamentos da Codificação de Textura intraquadros nos codificadores orientados por objeto, reconhecendo o estado da arte e propondo melhorias nos algoritmos de codificação existentes a partir da exploração das características de forma e textura do objeto a ser codificado. Também pretende-se identificar lacunas em pontos ainda não consolidados e apontar para perspectivas de trabalhos futuros. Dentre os objetivos específicos, incluem-se:

- uma análise comparativa dos algoritmos de particionamento da imagem em blocos, com relação à eficiência na redução do número de blocos a codificar e à complexidade de implementação, seguida de avaliação sobre a sua aplicabilidade em esquemas de codificação orientada por objeto;
- proposição de indicadores morfológicos que descrevam características de forma e textura dos objetos a serem codificados;
- utilização dos indicadores morfológicos como auxiliares na análise de eficiência dos algoritmos de codificação de textura orientados por objeto e na escolha dos melhores algoritmos;
- análise de estratégias para a escolha da direção preferencial de processamento para o algoritmo de extrapolação EI - *Extension/Interpolation*, a ser usado nos blocos de contorno em conjunto com a DCT baseada em blocos, e proposição de um esquema híbrido para a determinação dessa direção, tendo os indicadores morfológicos como elementos auxiliares;

– análise de estratégias para a seleção da primeira direção de processamento da SA-DCT e proposição de um esquema híbrido com esse objetivo, também baseado nos indicadores morfológicos;

– proposição de um codificador de textura orientado por objeto adaptativo, que seja função da taxa de bits, da forma e da textura do objeto segmentado.

1.4 Organização da Tese

Esta tese está organizada em oito capítulos. O presente capítulo fornece a motivação para o estudo da codificação de textura em codificadores orientados por objeto, descreve a evolução dos padrões de codificação de imagem e apresenta os objetivos e a organização da tese.

No Capítulo 2 é apresentado o Estado da Arte sobre codificação intraquadros na codificação de textura orientada por objeto através da DCT adaptativa à forma – SA-DCT – e da DCT baseada em blocos, que são as abordagens tratadas ao longo desta tese.

O Capítulo 3 trata de esquemas de particionamento adaptativo dos blocos da imagem - em contraste com o particionamento fixo convencional - visando à melhoria da codificação de objetos de forma arbitrária. Neste capítulo são propostos dois critérios de otimização para os esquemas de particionamento adaptativo dos blocos da imagem. Com base nesses critérios, é apresentada uma análise comparativa da eficiência dos diversos algoritmos para a redução do número de blocos de contorno e do número total de blocos a serem codificados, bem como à carga computacional exigida, apontando para os algoritmos que melhor atendam esse compromisso.

No Capítulo 4 são propostos alguns indicadores morfológicos que permitam descrever características de forma e de textura dos objetos a serem codificados. O objetivo inicial é sua utilização em uma análise preliminar de desempenho das duas abordagens de codificação de textura orientados por objeto considerados nesta tese: a DCT-2D (DCT baseada em blocos) associada aos algoritmos de extrapolação e a SA-DCT. O Capítulo 4 apresenta resultados de codificação para uma ampla faixa de taxa de bits, apontando para a escolha do algoritmo de extrapolação mais eficiente para o emprego conjunto com a DCT baseada em blocos. Em seguida, faz uma análise comparativa entre esse algoritmo e a SA-DCT, segundo os indicadores morfológicos propostos. Nesse capítulo também são detalhadas

as condições de realização dos experimentos, que serão mantidas por toda a tese.

O Capítulo 5 é dedicado a estratégias de escolha da direção preferencial de processamento (horizontal ou vertical) do primeiro conjunto de transformadas DCT-1D empregadas no algoritmo de extrapolação EI - *Extension/Interpolation* (EI-DCT). Inicialmente é considerada a questão de qual das estratégias baseadas nas variâncias dos comprimentos dos segmentos do objeto - MALV (*Maximum Lengths Variance*) ou MILV (*Minimum Lengths Variance*) - é mais adequada à determinação da direção preferencial para esse algoritmo de extrapolação, relacionando a eficiência dessas técnicas a um dos indicadores morfológicos. Para as situações em que não há uma boa correlação entre esse indicador morfológico e os desempenhos de MILV e MALV, é proposta uma nova estratégia para a determinação da primeira direção de processamento, que leva em consideração a capacidade de compactação de energia das transformadas nos coeficientes finais da DCT. Essa estratégia será denominada MACES (*MAximum Cumulative Energy Sum*). Finalmente, é proposto um esquema híbrido que seja capaz de escolher adaptativamente a primeira direção de processamento do algoritmo EI, tendo o indicador morfológico como a base de chaveamento entre as estratégias. Em seguida, é analisada a eficácia do emprego de esquemas de particionamento em blocos da imagem apresentados no Capítulo 3, antes da aplicação do algoritmo de extrapolação.

O Capítulo 6 trata de tópicos semelhantes aos considerados no Capítulo 5, porém aplicados no contexto da SA-DCT. Inicialmente é investigada a eficiência das estratégias MALV e MILV para a determinação da direção prioritária de processamento da SA-DCT, segundo o mesmo indicador morfológico. Com base nessa análise, é proposto um esquema híbrido que também chaveie adaptativamente entre as estratégias MALV, MILV e MACES, dependendo de qual é a mais adequada a grupos específicos de blocos de contorno. Esses grupos são definidos pelo número de pixels do bloco que efetivamente pertencem ao objeto. O uso de esquemas de particionamento da imagem associado à codificação por SA-DCT também é avaliado no Capítulo 6.

O Capítulo 7 tem por objetivo a proposição e análise de um codificador de textura orientado por objeto, que é adaptativo à taxa de bits, à forma e à textura do objeto segmentado. Dependendo desses parâmetros, o codificador faz o chaveamento entre o esquema híbrido da EI-DCT (proposto no Capítulo 5) e o esquema híbrido da SA-DCT (proposto no Capítulo 6). Considerando-se que a EI-DCT é, em geral, mais adequada à codificação

a taxas baixas, ao passo que a SA-DCT é indicada para a codificação a taxas mais altas, o codificador adaptativo buscará definir a estratégia mais adequada a um determinado grupo de blocos de contorno, de acordo com a taxa de bits e os indicadores de forma e textura dos objetos.

Finalmente, no Capítulo 8 são destacadas as principais contribuições desta tese, ao mesmo tempo em que são identificadas as lacunas e as possibilidades para estudos futuros, visando à continuidade do trabalho desenvolvido.

1.5 Lista de Símbolos

A seguir, apresenta-se a relação dos principais símbolos utilizados nesta tese:

- AE: Área Efetiva
- BBGM: Boundary-Block Group and Merging
- *bppo*: taxa de bits por pixel do objeto
- *bpcq*: taxa de bits por coeficiente a quantizar
- DCT: Discrete Cosine Transform
- DCT-1D: DCT unidimenional
- DCT-2D: DCT bidimenional
- DNPO : Distribuição de blocos de acordo com o Número de Pixels do Objeto
- EA: algoritmo de estrapolação Expanded Arbitrarily-Shaped
- EA-DCT: extrapolação do bloco de contorno através do algoritmo EA, seguido de codificação por DCT-2D
- EI: algoritmo de extrapolação Extension/Interpolation
- EI-DCT: extrapolação do bloco de contorno através do algoritmo EI, seguido de codificação por DCT-2D
- LPE: algoritmo de extrapolação Low-Pass-Extrapolation
- MACES: MAximum Cumulative Energy Sum
- MALV: MAximum Lengths Variance
- MILV: MInimum Lengths Variance
- NBC: Número de Blocos de Contorno
- NPO: Número de Pixels do Objeto

- NTB: Número Total de Blocos a codificar
- RPR: Razão Pico-Ruído (dB)
- SA-DCT: Shape-Adaptive DCT
  - NO-SA-DCT: SA-DCT não normalizada
  - PO-SA-DCT: SA-DCT pseudo-ortonormal
  - BBGM-SA-DCT: SA-DCT associada ao algoritmo BBGM
- SARP: Shape-Adaptive Region Partitioning
- SEA: algoritmo de estrapolação Simplified Expanded Arbitrarily-Shaped
- SEA-DCT: extrapolação do bloco de contorno através do algoritmo SEA, seguido de codificação por DCT-2D
- SEA-DCT-MPEG4: extrapolação do bloco de contorno através do algoritmo LPE, seguido de codificação por DCT-2D
- SPIHT: Set Partitioning In Hierarchical Treees
- TNPO : Texturas dos blocos de acordo com o Número de Pixels do Objeto
- TPR: Tempo de Processamento Relativo