



DISSERTAÇÃO DE MESTRADO EM ENGENHARIA  
ELÉTRICA

**REDUÇÃO DE COMPLEXIDADE  
NA COMPRESSÃO DE VÍDEO  
DE ALTA RESOLUÇÃO**

**Tiago Alves da Fonseca**

**Brasília, fevereiro de 2008**

**UNIVERSIDADE DE BRASÍLIA**

FACULDADE DE TECNOLOGIA



UNIVERSIDADE DE BRASÍLIA  
Faculdade de Tecnologia

DISSERTAÇÃO DE MESTRADO EM ENGENHARIA  
ELÉTRICA

**REDUÇÃO DE COMPLEXIDADE  
NA COMPRESSÃO DE VÍDEO  
DE ALTA RESOLUÇÃO**

**Tiago Alves da Fonseca**

Dissertação de mestrado submetida ao Departamento de Engenharia Elétrica da Faculdade de Tecnologia da Universidade de Brasília, como parte dos requisitos necessários para a obtenção do grau de mestre.

Banca Examinadora

Prof. Ricardo Lopes de Queiroz, PhD. UnB/ ENE (Orientador) \_\_\_\_\_

Prof. Juliana Fernandes Camapum, PhD. UnB/ ENE (Examinador Interno) \_\_\_\_\_

Prof. Eduardo Antônio Barros da Silva, PhD. UFRJ/ COPPE (Examinador Externo) \_\_\_\_\_

## FICHA CATALOGRÁFICA

FONSECA, TIAGO ALVES

Redução de Complexidade na Compressão  
de Vídeo de Alta Resolução. [Distrito Federal] 2008.

xii, 78p., 297 mm (ENE/FT/UnB, Mestre, Telecomunicações  
Processamento de Sinais, 2008). Dissertação de Mestrado.  
Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Elétrica.

- |                            |                    |
|----------------------------|--------------------|
| 1. Compressão de vídeo     | 2. Complexidade    |
| 3. Vídeo de alta resolução | 4. DPCM            |
| 5. H.264                   | 6. MPEG4           |
| I. ENE/FT/UnB              | II. Título (série) |

## REFERÊNCIA BIBLIOGRÁFICA

FONSECA, T. A. da (2008). Redução de Complexidade na Compressão de Vídeo de Alta Resolução. Dissertação de Mestrado em Engenharia Elétrica com ênfase em Telecomunicações, Publicação PPGENE.DM - 323/08, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 78p.

## CESSÃO DE DIREITOS

NOME DO AUTOR: Tiago Alves da Fonseca.

TÍTULO DA DISSERTAÇÃO DE MESTRADO: Redução de Complexidade na Compressão de Vídeo de Alta Resolução.

GRAU / ANO: Mestre / 2008

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta dissertação de mestrado pode ser reproduzida sem a autorização por escrito do autor.

---

Tiago Alves da Fonseca  
SQN 114, Bloco G, Ap. 610  
70.764-070 Brasília - DF - Brasil.

## **Dedicatória**

*A meus pais, que cedo mostraram-me a importância da formação e sempre lutaram para me proporcionar uma educação de qualidade. À minha mãe Eliete e minha irmã Vanessa, que têm de “aturar” meu humor e minha “ausência” todos os dias. À minha família, pelos momentos de diversão e minha ausência. A Juliana, pelos momentos agradáveis de companheirismo e consolação. Aos amigos, pela motivação e compreensão.*

*Tiago Alves da Fonseca*

*“Não queiras ser como aquele catavento dourado do grande edifício; por muito que brilhe e por mais alto que esteja, não conta para a solidez da obra.*

*— Oxalá sejas como um velho silhar oculto nos alicerces, debaixo da terra, onde ninguém te veja; por ti não desabará a casa.”*

Josemaría Escrivá, Caminho

## **Agradecimentos**

*Agradeço à minha família, por me incurrir a importância da formação desde cedo, especialmente a meus pais (Eliete, Valdir José, Afonso Celso e Maria da Conceição (in memoriam)) com cuja convivência puderam me passar experiências de suas vidas e sempre me orientaram para o caminho correto.*

*Aos amigos, pelos momentos descontraídos e pelo companheirismo.*

*À turma do GPDS (não citarei nomes para não correr o risco de esquecer algum de vocês) pelo companheirismo, apoio e motivação para minha vida acadêmica.*

*Ao conselho diretor da MUX Tecnologia, pela preocupação constante com a formação de seu corpo técnico. ;D*

*Ao amigo Edson Mintsu Hung, outro que tem me aturado praticamente 24h/7dias.*

*Aos amigos revisores (Alexandre Zaghetto, Bruno Macchiavello e Mintsu), pelas sugestões e apoio à composição deste trabalho.*

*Ao Departamento de Engenharia Elétrica da Universidade de Brasília que, juntamente ao Centro de Desenvolvimento Tecnológico e a Hewlett Packard do Brasil, proporcionaram recursos e ambiente favorável à formação acadêmica.*

*Por fim, ao meu professor orientador Ricardo Lopes de Queiroz, pela oportunidade de trabalho, pelo compartilhamento de seu conhecimento e motivação à pesquisa.*

*Tiago Alves da Fonseca*

---

## RESUMO

O H.264/AVC é o mais novo padrão de compressão de vídeo e é tomado como estado da arte. Ele proporciona melhorias consideráveis de desempenho quando comparado a outros padrões existentes. Entretanto, como outros padrões, ele é um codificador híbrido composto por módulo preditivo e de transformada. No presente trabalho, propomos duas abordagens diferentes para implementação do estágio preditivo. A primeira idéia é usar dados originais em vez de dados reconstruídos para realizar os testes de predição na escolha do melhor modo de predição. O resíduo, todavia, continua sendo calculado usando dados anteriormente decodificados de forma a evitar *drifting*; essa técnica permite a paralelização do estágio de predição Inter-quadros, a operação mais demorada no H.264/AVC. A segunda contribuição reduz a complexidade da codificação pela supressão dos testes de predição de modos menos frequentes. Resultados mostram que, para seqüências de alta resolução, as metodologias propostas implicam pouca perda de qualidade no sinal aliada a uma grande economia de recursos.

---

## ABSTRACT

H.264/AVC is the newest, state-of-art, video compression standard. It leads to substantial performance improvement compared to other existing standards. However, like other video standards, it is a hybrid predictive-transform coder. In this work, we propose two different approaches to implement the prediction stage. The first idea is to employ original data rather than reconstructed ones to perform prediction tests before choosing the best mode. The residue, however, is evaluated using previously decoded data in order to avoid drifting. The technique allows parallelization of the inter-prediction stage, which is the most time consuming operation in H.264/AVC. The second contribution reduces the overall encoder complexity by avoiding less frequent prediction mode tests. Results show that, for high definition sequences, the proposed methodologies introduced very small quality losses associated with large reduction of computational burden.



# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
1.1	CONTEXTUALIZAÇÃO	1
1.2	DEFINIÇÃO DO PROBLEMA	3
1.3	OBJETIVOS	5
1.4	APRESENTAÇÃO DO MANUSCRITO	6
<b>2</b>	<b>COMPRESSÃO DE IMAGENS E VÍDEO</b>	<b>7</b>
2.1	INTRODUÇÃO	7
2.2	CONCEITOS BÁSICOS DE PROCESSAMENTO DE VÍDEO DIGITAL	7
2.2.1	AQUISIÇÃO DE SINAIS DE VÍDEO DIGITAL	7
2.2.2	CODIFICAÇÃO DE CORES	8
2.2.3	MÉTRICAS DE AVALIAÇÃO DA QUALIDADE DE SINAIS CODIFICADOS	9
2.3	TÉCNICAS DE COMPRESSÃO DE VÍDEO	9
2.3.1	A TRANSFORMADA DISCRETA DE COSSENO (DCT): VIABILIZAÇÃO DA COMPRESSÃO	10
2.3.2	QUANTIZAÇÃO E CODIFICAÇÃO DE ENTROPIA: REALIZAÇÃO DA COMPRESSÃO	10
2.3.3	CODIFICADOR DE VÍDEO DPCM HÍBRIDO	13
2.4	O CODIFICADOR H.264/AVC	20
2.4.1	PREDIÇÃO INTRA-QUADRO	20
2.4.2	PREDIÇÃO INTER-QUADROS	22
2.4.3	CODIFICAÇÃO POR TRANSFORMADAS	26
2.4.4	CODIFICAÇÃO DE ENTROPIA	26
2.4.5	FILTRO DE <i>Deblocking</i> ADAPTATIVO	27
2.4.6	PERFIS DO H.264/AVC	27
<b>3</b>	<b>AVALIAÇÃO DE COMPLEXIDADE DO CODIFICADOR H.264/AVC</b>	<b>31</b>
3.1	INTRODUÇÃO	31
3.2	PERFIL DE COMPLEXIDADE DO CODIFICADOR H.264/AVC <i>High Profile</i>	32
3.2.1	PREDIÇÃO COM O LAÇO ABERTO	33
3.3	POLARIZAÇÃO DOS MODOS DE PREDIÇÃO	34
<b>4</b>	<b>PREDIÇÃO COM LAÇO ABERTO</b>	<b>41</b>
4.1	DESCRIÇÃO	41
4.1.1	PREDIÇÃO DE MACROBLOCOS NO H.264/AVC	41
4.1.2	PREDIÇÃO BASEADA EM DADOS ORIGINAIS	42
4.2	RESULTADOS EXPERIMENTAIS	51
4.3	DISCUSSÃO	55
<b>5</b>	<b>MÓDULO ANALISADOR ESTATÍSTICO PARA REDUÇÃO DE COMPLEXIDADE DE PREDIÇÃO</b>	<b>57</b>
5.1	DESCRIÇÃO	57
5.1.1	AVALIAÇÃO DO TAMANHO DO SUBCONJUNTO DE MACROBLOCOS PARA DETERMINAÇÃO DOS MODOS DOMINANTES DE UM QUADRO	58
5.1.2	AVALIAÇÃO DO ERRO MÉDIO NA PREDIÇÃO DOS MODOS DOMINANTES DO QUADRO ATUAL A PARTIR DO SUBCONJUNTO AMOSTRAL	60
5.1.3	AVALIAÇÃO DO ERRO MÉDIO NA PREDIÇÃO DOS MODOS DOMINANTES DO PRÓXIMO QUADRO A PARTIR DO SUBCONJUNTO AMOSTRAL	63

5.1.4	REDUTOR DE COMPLEXIDADE .....	66
5.2	RESULTADOS .....	66
5.3	DISCUSSÃO .....	71
<b>6</b>	<b>CONCLUSÕES.....</b>	<b>73</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>74</b>

# LISTA DE FIGURAS

1.1	Quadros consecutivos da seqüência Foreman.....	3
2.1	Bases da DCT para blocos de $8 \times 8$ <i>pixels</i> . Adaptado de [1]. .....	11
2.2	Dois quadros consecutivos da seqüência Foreman. ....	12
2.3	Fluxo óptico composto por vetores de movimento para cada bloco de $8 \times 8$ <i>pixels</i> da imagem da Figura 2.2(b) com relação à Figura 2.2(a).....	14
2.4	Versões do quadro da Figura 2.2(b): (a) original e (b) compensado com relação à Figura 2.2(a) empregando o fluxo óptico da Figura 2.3. ....	15
2.5	Resíduo entre os quadros consecutivos (a) sem predição e (b) utilizando predição por compensação de movimentos.....	16
2.6	Codificador de vídeo DPCM híbrido. Adaptado de [2]. ....	17
2.7	Decodificador de vídeo DPCM híbrido. Adaptado de [2]. ....	19
2.8	Janela causal (a) para codificação e modos de predição para codificação Intra-quadro com suas direções de interpolação planar para blocos $4 \times 4$ e $8 \times 8$ (b) e $16 \times 16$ (c). O modo 2, não apresentado em (b), é o DC.....	21
2.9	Partições de macroblocos. ....	22
2.10	Organização das partições em uma estrutura hierárquica. Adaptado de [1]. ....	23
2.11	Compensação de movimento com múltiplos quadros de referência em quadros P. ....	23
2.12	Compensação de movimento com múltiplos quadros de referência em quadros B.....	24
2.13	Quadros de compressão de vídeo ordenado de acordo com a seqüência de exibição. Adaptado de [1]. ....	25
2.14	Quadros de compressão de vídeo ordenado de acordo com a seqüência de compressão. Adaptado de [1]. ....	25
2.15	Ilustração dos perfis do H.264. Adaptado de [1]. ....	28
3.1	Frequência de ocorrência de modos de predição $\times$ resolução para a seqüência Pedestrian Area.....	35
3.2	Frequência de ocorrência de modos de predição $\times$ resolução para a seqüência Riverbed. ....	35
3.3	Frequência de ocorrência de modos de predição $\times$ resolução para a seqüência Rushhour. ....	36
3.4	Frequência de ocorrência de modos de predição $\times$ resolução para a seqüência Sunflower....	36
3.5	Frequência de ocorrência de modos de predição $\times$ QP para a seqüência Pedestrian Area. ....	37
3.6	Frequência de ocorrência de modos de predição $\times$ QP para a seqüência Riverbed. ....	38
3.7	Frequência de ocorrência de modos de predição $\times$ QP para a seqüência Rushhour.....	38
3.8	Frequência de ocorrência de modos de predição $\times$ QP para a seqüência Sunflower. ....	39
4.1	Codificador de Vídeo DPCM Híbrido. Adaptado de [2]. ....	41
4.2	Processo de predição Inter-quadros (a) paralelizado e nova estrutura do codificador (b) para predição paralelizada. ....	43
4.3	Diagrama de predição original com cinco quadros de referência. ....	44
4.4	Ilustração do processo de predição original ( (a) a (i) ) para cinco quadros de referência. ....	47
4.5	Diagrama de predição proposto com cinco quadros de referência. ....	48
4.6	Processo de predição proposto ( (a) a (e) ) para cinco quadros de referência. ....	50
4.7	Curva de taxa $\times$ distorção para a seqüência Pedestrian Area: JM10.2 $\times$ JM10.2-OL.....	52
4.8	Curva de taxa $\times$ distorção para a seqüência Rushhour: JM10.2 $\times$ JM10.2-OL.....	52
4.9	Curva de taxa $\times$ distorção para a seqüência Riverbed: JM10.2 $\times$ JM10.2-OL. ....	53
4.10	Curva de taxa $\times$ distorção para a seqüência Sunflower: JM10.2 $\times$ JM10.2-OL.....	53
4.11	Curva de taxa $\times$ distorção para a seqüência CIF Mobile: JM10.2 $\times$ JM10.2-OL.....	54
4.12	Curva de taxa $\times$ distorção para a seqüência CIF Foreman: JM10.2 $\times$ JM10.2-OL.....	54

4.13	Curva de taxa × distorção para a seqüência CIF Coastguard: JM10.2 × JM10.2-OL.....	55
5.1	Fluxograma da metodologia de determinação dos modos de predição dominantes. ....	58
5.2	Variação do tamanho da amostra <i>versus</i> proporção da população a ser verificada.....	60
5.3	Taxa de erro médio na predição dos modos de predição dominantes do quadro atual × percentagem de macroblocos onde são avaliados todos os modos para a seqüência Pedestrian Area. ....	61
5.4	Taxa de erro médio na predição dos modos de predição dominantes do quadro atual × percentagem de macroblocos onde são avaliados todos os modos para a seqüência Riverbed. ....	62
5.5	Taxa de erro médio na predição dos modos de predição dominantes do quadro atual × percentagem de macroblocos onde são avaliados todos os modos para a seqüência Rushhour. ....	62
5.6	Taxa de erro médio na predição dos modos de predição dominantes do quadro atual × percentagem de macroblocos onde são avaliados todos os modos para a seqüência Sunflower. ....	63
5.7	Taxa de erro médio na predição dos modos de predição dominantes do próximo quadro × percentagem de macroblocos onde são avaliados todos os modos para a seqüência Pedestrian Area. ....	64
5.8	Taxa de erro médio na predição dos modos de predição dominantes do próximo quadro × percentagem de macroblocos onde são avaliados todos os modos para a seqüência Riverbed. ....	64
5.9	Taxa de erro médio na predição dos modos de predição dominantes do próximo quadro × percentagem de macroblocos onde são avaliados todos os modos para a seqüência Rushhour. ....	65
5.10	Taxa de erro médio na predição dos modos de predição dominantes do próximo quadro × percentagem de macroblocos onde são avaliados todos os modos para a seqüência Sunflower. ....	65
5.11	Curva de taxa × distorção para a seqüência Pedestrian Area: JM10.2 × JM10.2-ANA.....	67
5.12	Curva de taxa × distorção para a seqüência Riverbed: JM10.2 × JM10.2-ANA.....	67
5.13	Curva de taxa × distorção para a seqüência Rushhour: JM10.2 × JM10.2-ANA.....	68
5.14	Curva de taxa × distorção para a seqüência Sunflower: JM10.2 × JM10.2-ANA.....	68
5.15	Curva do perfil de redução de complexidade (quanto tempo foi economizado) para as quatro seqüências de vídeo analisadas.....	69
5.16	Curva de taxa × distorção para a seqüência Mobile: JM10.2 × JM10.2-ANA.....	69
5.17	Curva de taxa × distorção para a seqüência Foreman: JM10.2 × JM10.2-ANA.....	70
5.18	Curva de taxa × distorção para a seqüência Coastguard: JM10.2 × JM10.2-ANA.....	70

## LISTA DE TABELAS

3.1	Complexidade computacional relativa para codificação somente Intra-quadro da seqüência HD Pedestrian Area pelo codificador H.264/AVC <i>High Profile</i> com otimização taxa $\times$ distorção ativa. ....	32
3.2	Complexidade computacional relativa para codificação da seqüência HD Pedestrian Area pelo codificador H.264/AVC <i>High Profile</i> com otimização taxa $\times$ distorção ativa e variados tamanhos de janela de busca para estimação de movimentos. ....	33



# LISTA DE SIGLAS, ABREVIACOES E ACRONIMOS

## Abreviaoes, Acronimos e Siglas

AVC	<i>Advanced Video Coding</i>
CABAC	<i>Context-Based Adaptive Binary Arithmetic Coding</i>
CAVLC	<i>Context-Adaptive Variable Length Coding</i>
CD	<i>Compact Disk</i>
DVD	<i>Digital Versatile Disk</i>
FRExt	<i>Fidelity Range Extensions</i>
GB	Giga bytes (1073741824 bytes)
HD	<i>High Definition</i>
IEC	<i>International Electrotechnical Commission</i>
ISO	<i>International Standards Organization</i>
ITU	<i>International Telecommunication Union</i>
JPEG	<i>Joint Photographic Experts Group</i>
JVT	<i>Joint Video Team</i>
MPEG	<i>Motion Picture Experts Group</i>
MSE	<i>Mean Square Error</i>
PSNR	<i>Peak Signal to Noise Ratio</i>
QP	<i>Quantization Parameter</i>
SD	<i>Standard Definition</i>
VCEG	<i>Video Coding Experts Group</i>
VHS	<i>Video Home System</i>
VLC	<i>Variable Length Code</i>



# 1 INTRODUÇÃO

## 1.1 CONTEXTUALIZAÇÃO

O avanço e a popularização da tecnologia digital foi notável na últimas duas décadas. Canais de comunicação tiveram, e ainda têm, suas capacidades de transmissão aumentadas constantemente, a disponibilidade de canais de banda larga em domicílios já é uma realidade e meios de armazenamento de informações digitais em massa têm sua capacidade de armazenamento superada a cada mês. Um olhar superficial sobre esses avanços nos leva a acreditar que não se justifica o uso de técnicas de compressão de sinais. [2]

Contudo, uma análise mais detalhada nos revela que até mesmo o DVD (*Digital Versatile Video*), um dos meios mais difundidos e de grande capacidade de armazenamento de dados digitais, seria inútil sem o auxílio de compressão. Tomemos como exemplo o armazenamento de um vídeo de definição padrão (SD, *standard definition*) com quadro de dimensões de 720 *pixels* (pontos elementares de imagem) de largura por 480 *pixels* de altura e capturado a uma taxa de 30 quadros por segundo. Para armazenar um quadro de vídeo em cores sem compressão, gastariam-se 518.400 bytes; como a taxa de aquisição é de 30 quadros por segundo, para armazenar um segundo gastariam-se 15.552.000 bytes. Dada a capacidade de armazenamento de um DVD, 4,7 GB ( $4,7 \times 2^{30}$  bytes), é fácil concluir que seria possível armazenar apenas 5,4 minutos de vídeo sem compressão. Sabe-se que um DVD é capaz de armazenar não só vídeos de mais de duas horas, mas também legendas, áudios e outros conteúdos extras usados na promoção de filmes comerciais. Nota-se, então, a necessidade da compressão de vídeo para seu armazenamento.

Transmissão de vídeo em dispositivos móveis é um campo que traz novos desafios para a indústria de telecomunicações. As restrições que envolvem esse tipo de dispositivo são inúmeras e vão desde limitado espectro de frequências de transmissão (esse é um recurso escasso e que não pode ser aumentado) a poder de processamento restrito e recursos finitos de energia. Somente as técnicas mais modernas de codificação de sinais garantirão que tais dispositivos consigam capturar e transmitir vídeo em tempo real, com a maior qualidade possível, usando a menor largura de banda e pela maior quantidade de tempo. Logo, a compressão de vídeo se faz necessária também para a transmissão de vídeo.

Portanto, mesmo com avanços constantes nas capacidades de armazenamento e transmissão de dados digitais, a compressão ainda será um componente essencial para serviços de comunicação multimeios por

muitos anos.

A compressão de um **sinal** (denominação dada a um ente que carrega **informação** [3]) pode ser encarada como um processo que resulta numa representação mais eficiente que trata melhor as informações redundantes que o mesmo carrega. São duas as classes principais de técnicas que podem ser empregadas nesse processo: as técnicas de compressão sem perdas e as de compressão com perdas.

Em um sistema de compressão sem perdas, a redundância estatística é removida de forma que o sinal original possa ser reconstruído perfeitamente no receptor. O princípio básico da compressão sem perdas é gastar a menor quantidade de bits (unidade básica de informação digital) possível para codificar um sinal sem perder informação. Logo, dado um alfabeto de símbolos usados na representação de uma certa informação, procura-se expressar os símbolos mais freqüentes com menos bits do que os usados nos símbolos mais raros. Isso acaba resultando numa redução da quantidade de bits total usada para expressar o sinal ao se comparar com caso em que cada símbolo é codificado com a mesma quantidade de bits. Esses métodos são extremamente importantes quando um erro na versão comprimida do sinal pode comprometer a integridade da informação, como é o caso de documentos eletrônicos, registros bancários etc. Contudo, apresentam desempenhos modestos quando aplicados diretamente a imagens ou vídeos digitais. [4]

Por outro lado, há certos tipos de informações em que falhas na reconstrução exata não constituem um problema. Na codificação de sinais de áudio, o valor exato da amostra digitalizada não é necessário para garantir a inteligibilidade e certas quantidades de informação podem ser descartadas a depender do contexto. Por exemplo, não se toleram grandes perdas na codificação de uma música em qualidade de CD (*Compact Disk*), mas se o sinal é voz que vai ser transmitida por canal telefônico, aceita-se perda significativa da informação sem prejudicar a comunicação.

As técnicas de codificação baseadas no paradigma da compressão com perdas implicam grande compressão, a qual é atingida às custas de perda de qualidade do sinal tratado cuja versão decodificada agora passa a ser diferente da versão original; à diferença resultante do processo de codificação do sinal dá-se o nome de **distorção**. A comunicação de sinais de vídeo, por demandar a transmissão de grandes quantidades de informação, baseia-se notadamente em técnicas de compressão com perdas.

## 1.2 DEFINIÇÃO DO PROBLEMA

A codificação de vídeo é caracterizada pelo processo de compressão e descompressão de um sinal de vídeo digital. A compressão pode ser entendida como um processo que procura uma representação mais eficiente do sinal. Uma observação superficial de um sinal de vídeo mostra que há muita redundância entre quadros consecutivos de uma seqüência de vídeo (veja a Figura 1.1).



Figura 1.1: Quadros consecutivos da seqüência Foreman.

Note que boa parte do conteúdo visual da cena permanece constante entre um quadro e outro. Uma estratégia inicial para compressão poderia ser mandar somente as diferenças existentes entre os quadros. Convencionou-se chamar de codificação **Inter-quadros** [4, 5] as técnicas existentes na literatura que abordam a redundância temporal do sinal de vídeo.

Mas, como será mandado o primeiro dos quadros de uma seqüência de vídeo digital? Usar um algoritmo de compressão de imagens é uma solução muito comum para esse tipo de problema. As estratégias mais difundidas para extração de redundâncias espaciais em uma imagem digital baseiam-se na aplicação de uma operação matemática chamada de **transformada**. O princípio de funcionamento de uma transformada é simples: representar o sinal de uma maneira mais conveniente a uma dada situação; em compressão, os sinais são transformados de forma a “compactar” a **energia**. O emprego exclusivo de técnicas de codificação de imagens em que não são feitas referências a outros quadros na compressão de um quadro denomina-se de codificação **Intra-quadro**. [4, 5]

Um codificador de vídeo é usualmente composto pela agregação de técnicas de codificação Intra-quadro e Inter-quadros. À primeira vista, o projeto de um codificador de vídeo pode parecer preocupado somente com a redução das demandas de armazenamento ou de largura de banda em transmissões de

sinais de vídeo; todavia, envolvem-se outros aspectos relativos ao desempenho do sistema de codificação. Parâmetros como critérios de taxa  $\times$  distorção, complexidade do algoritmo utilizado, características do canal de transmissão, estatísticas da fonte de vídeo, entre outros, devem ser levados em consideração na seleção das técnicas de codificação mais convenientes. Como são inúmeras as possibilidades para a construção de um codificador, há a necessidade de estabelecer padrões de compressão de vídeo para promover a interoperabilidade entre diferentes aplicações.

Seguindo uma breve linha cronológica, o H.261 [5], desenvolvido pela ITU-T (*International Telecommunication Union, Telecommunication Standardization Sector*), foi o primeiro padrão de codificação de vídeo difundido com sucesso, capaz de comprimir vídeo para as taxas de transmissão mais comuns na época (entre 80 e 320 kbit/s para comunicações em vídeo digital). Sua estrutura ainda é herdada por muitos codificadores modernos. Em seguida, surge o MPEG-1 [6], desenvolvido conjuntamente pela ISO (*International Organization for Standardization*) e a IEC (*International Electrotechnical Commission*), codificador bem difundido e com faixa de operação maior que o padrão anterior (1 a 2 Mbit/s) e capaz de proporcionar qualidades superiores às de fitas VHS (*Video Home System*) consumindo 1,5 Mbit/s de taxa de transmissão. O MPEG-2 [7], seu sucessor, foi desenvolvido de forma a superar o desempenho de padrões anteriores. Ele se destaca por promover um avanço maior na qualidade de imagem e por sua popularidade: é o responsável por impulsionar a difusão de sinais de televisão digital em resolução padrão e alta definição. [8]

Com foco em transmissão de vídeo em baixas taxas, o H.263 [9] foi considerado o estado da arte da sua época. Como sucessor do H.261, sua faixa de operação era inicialmente em torno de 10 a 30 kbit/s, mas ela foi estendida para 10 a 2048 kbit/s. Retornando à família de codificadores MPEG, surge o MPEG-4 [10], criado com objetivo de padronizar os métodos de codificação mais eficientes e mais genéricos disponíveis até o momento de sua proposição, capazes de manipular vários tipos de dados audiovisuais; seu ponto de partida é o codificador H.263 no perfil *Baseline*. Isso significa que decodificadores compatíveis com MPEG-4 devem ser capazes de decodificar vídeos comprimidos com H.263 *Baseline*. [11]

O H.264/AVC (*Advanced Video Coding*) [12] é o mais novo padrão de compressão de vídeo e resulta da colaboração entre as equipes *ISO/IEC Moving Picture Experts Group* (MPEG) e *ITU-T Video Coding Experts Group* (VCEG) sobre a agregação denominada JVT (*Joint Video Team*). Além de promover a desejada interoperabilidade, os objetivos dessa padronização foram oferecer elevada eficiência de compressão de uma classe específica de sinais de vídeo, aqueles resultantes da captura de cenas reais (e não sintéticas), além de proporcionar representação de vídeo conveniente à transmissão tanto para aplicações

interativas quanto para não-interativas. Os ganhos de compressão trazidos por esse novo padrão vieram, de certa maneira, às custas de elevada complexidade computacional. A título de ilustração, a complexidade computacional de seu decodificador é quatro vezes maior que a do MPEG-2 e duas vezes maior que a do MPEG-4 *Visual*. [13]

O estado da arte em compressão de vídeo atingido pelo H.264/AVC foi em grande parte resultado do refinamento de técnicas de codificação aplicadas a outros padrões. Como destaque, podemos elencar:

- Estágio preditivo variado composto por compensação de movimentos com variadas partições, flexibilidade no emprego de quadros de referência e vetores de movimento com precisão refinada de até 1/4 de *pixel* e predição Intra-quadro com variadas partições.
- Módulo de transformada mais granular por empregar transformadas inteiras reversíveis de tamanhos  $2 \times 2$ ,  $4 \times 4$  e  $8 \times 8$ , aproveitando os variados tamanhos de partições.
- Codificador de entropia mais eficiente e que emprega contextos adaptativos na codificação dos elementos sintáticos.

A aplicação dos refinamentos listados acima resultam aumento de complexidade tanto no codificador, como no decodificador.[13, 14]

### 1.3 OBJETIVOS

Quando o cenário é a compressão de sinais de vídeo de alta definição (HD, *High Definition*), a complexidade computacional torna-se um problema evidente, uma vez que o volume de dados que deve ser analisado pelo codificador é elevado e a codificação de vídeo em tempo real passa a ser um desafio.

Grande parte da complexidade computacional do codificador H.264/AVC está concentrada em seu estágio preditivo. [14] Portanto, uma abordagem para viabilizar a codificação de vídeos HD em tempo real é alterar sua implementação em tópicos não-normativos, não necessitando modificar o decodificador. O emprego de **estimação de movimentos** subótimas é uma das soluções mais tradicionais nesse contexto: evitam-se testes desnecessários na varredura da janela de busca pelo melhor casamento. Para a viabilização de comunicações sem fio e em tempo real, onde canal e energia são recursos escassos, usa-se o conceito de escalabilidade em complexidade e generaliza-se a otimização taxa  $\times$  distorção para taxa  $\times$  distorção  $\times$  complexidade. [15] Enquanto o sistema tem bastante energia, ele emprega todo o esforço em comprimir a seqüência de vídeo com qualidade máxima. À medida em que a energia torna-se um bem escasso,

o codificador codifica a seqüência de vídeo empregando menos complexidade e resultando em menor qualidade mas garantindo a comunicação do vídeo pelo maior tempo possível.

O presente trabalho propõe novas metodologias para a manipulação da complexidade do estágio preditivo do H.264/AVC. Uma das técnicas sugere uma quebra no paradigma de codificação DPCM híbrido por meio da paralelização do módulo de predições, estrutura notadamente seqüencial. O trabalho de várias máquinas de predição operando paralelamente pode viabilizar a compressão de vídeo HD em tempo real.

A análise de estatísticas dos **modos de predição** utilizados pela codificação de uma seqüência de vídeo HD dá-nos indícios de que testes de predição podem ser melhor aplicados se modos de predição menos prováveis forem descartados. É comum verificar seqüências de vídeo HD em que mais de 50% dos **macroblocos** são codificados usando apenas um ou dois dos possíveis modos; descobrir os modos dominantes e empregá-los preferencialmente proporciona uma codificação mais rápida com pequenas perdas em qualidade. Um analisador estatístico dos modos é proposto nessa dissertação como alternativa na redução da complexidade do codificador.

## 1.4 APRESENTAÇÃO DO MANUSCRITO

No Capítulo 2 apresentam-se alguns conceitos envolvidos em compressão de vídeo e o codificador H.264/AVC é discutido com maiores detalhes. No Capítulo 3 são apresentadas as motivações para o presente trabalho por meio do detalhamento do problema a ser tratado. Em seguida, no Capítulo 4, apresentamos a quebra do laço de predição como primeira contribuição na viabilização da codificação de vídeos HD. A descrição do analisador estatístico e seu desempenho é deixada para o Capítulo 5. O trabalho é finalizado com as Conclusões, no Capítulo 6.

## 2 COMPRESSÃO DE IMAGENS E VÍDEO

### 2.1 INTRODUÇÃO

A compressão de um sinal pode ser encarada como uma representação mais eficaz que trata melhor as informações redundantes que o mesmo carrega. A fim de alcançarmos-la, é possível optar por duas classes de técnicas: as técnicas de compressão sem perdas e as de compressão com perdas. Em um sistema de compressão sem perdas, a redundância estatística é removida de forma que o sinal original possa ser reconstruído perfeitamente no receptor. Contudo, esses métodos apresentam desempenhos modestos quando aplicados diretamente a sinais de imagens ou vídeo. As técnicas de compressão de vídeo difundidas são baseadas no paradigma da compressão com perdas, em que grande compressão é atingida às custas de perda de qualidade do sinal tratado, cuja versão decodificada agora passa a ser diferente, mas de certa maneira fiel à versão original. Nesse conjunto de técnicas, quanto maior a degradação inserida no sinal, menor será a versão comprimida; o desafio de um algoritmo de compressão de vídeo é comprimir eficientemente minimizando a distorção inerente ao processo.

### 2.2 CONCEITOS BÁSICOS DE PROCESSAMENTO DE VÍDEO DIGITAL

Antes de iniciar a apresentação do processo de compressão de um sinal de vídeo digital, é necessário definir alguns conceitos.

#### 2.2.1 Aquisição de sinais de vídeo digital

Um sinal de vídeo digital é composto por uma seqüência de imagens digitais (quadros) capturadas de uma cena real ou geradas sinteticamente. Uma imagem digital de uma cena é resultado de transformação de um sinal bidimensional de parâmetros contínuos do mundo real para uma versão de parâmetros discretos e de amplitudes quantizadas, manipuláveis por sistemas digitais. [16, 17]

O processo de transformar parâmetros contínuos em parâmetros discretos é definido como amostragem. Mediante amostragem espacial, a cena bidimensional composta por infinitos pontos passa a ser representada por uma matriz retangular finita de pontos. O processo que limita o domínio dos valores de intensidade para cada um dos pontos amostrados é chamado de quantização. [18]

A fim de adquirir adequadamente a seqüência de vídeo, ainda é necessária a realização de amostragem no domínio do tempo que proporcione a mesma sensação de continuidade temporal existente no mundo real. Assim, o vídeo digital fica representado por amostras espaço-temporais cuja intensidade é composta por um conjunto números que representam o brilho (luminância) e a cor (crominância) da amostra.

### 2.2.2 Codificação de cores

A representação de cenas reais por imagens digitais leva em consideração o conteúdo de cores presente na cena, que será posteriormente mostrada em um meio de apresentação digital. Enquanto imagens monocromáticas necessitam apenas de um valor por amostra espacial para representar o conteúdo do sinal, no caso composto somente por brilho ou luminância, imagens coloridas requerem mais dois valores para representar as informações de crominância.

A maneira mais tradicional de representar uma imagem colorida é por meio do uso do espaço de cores RGB, no qual as informações de crominância e luminância de uma amostra do sinal são codificadas levando em consideração que cores visíveis podem ser representadas aproximadamente por ponderações de três componentes primárias de cor: vermelho (*Red*), verde (*Green*) e azul (*Blue*). [19] Logo para cada elemento/amostra da imagem (*pixel*), são necessários três valores para armazenar adequadamente as informações de cores. A simplicidade na captura das cores e na geração das mesmas por dispositivos eletrônicos justifica a popularidade deste tipo de codificação.

Por outro lado, o fato de o sistema RGB não levar em conta detalhes do comportamento do sistema visual humano deixa-o em desvantagem em relação a outros espaços de cores. Sabe-se que o olho humano é menos sensível a informações de crominância que de luminância. [20] O espaço de cores YCbCr usa essa característica como vantagem, tornando-se mais eficiente ao representar imagens coloridas. Este codifica informações de cores separadamente das informações de luminância, que necessitam de uma maior resolução. Para a determinação das suas componentes, usam-se as seguintes expressões:

$$\begin{aligned}
 Y &= k_r R + (1 - k_b - k_r)G + k_b B \\
 C_b &= \frac{0.5}{1 - k_b}(B - Y) \\
 C_r &= \frac{0.5}{1 - k_r}(R - Y)
 \end{aligned} \tag{2.1}$$

onde  $k$  são fatores de ponderação [21]. Um dos melhores atributos desse espaço de cores é que as componentes  $C_r$  e  $C_b$  podem ser representadas em resolução menor que  $Y$  devido à supracitada característica do sistema visual humano. Isso implica redução da quantidade de dados necessária para

codificar cores sem perdas de qualidade visual, por si só uma etapa de compressão do sinal.

### 2.2.3 Métricas de avaliação da qualidade de sinais codificados

Quantificar a qualidade visual é um problema difícil e impreciso dada a gama de fatores que podem interferir nos resultados. Numa tentativa de dispor de um método objetivo e de baixo custo computacional foram desenvolvidas algumas métricas em cujo cálculo não é levada em consideração a interação com observadores. [2]

A métrica mais popular para avaliação da qualidade de vídeos codificados é a *PSNR* (*Peak Signal to Noise Ratio*), medida em escala logarítmica e calculada a partir da razão entre o quadrado da maior intensidade que o sinal pode assumir (usualmente  $(2^n - 1)^2$ , onde  $n$  é o número de bits usados para codificar uma amostra do sinal) e o erro quadrático médio (*MSE*) do sinal original para o sinal codificado.

$$PSNR_{dB} = 10 \log_{10} \frac{(2^n - 1)^2}{MSE} \quad (2.2)$$

Sua simplicidade algorítmica é um dos fatores que a tornaram extremamente atrativa, todavia deve-se levar em conta suas vulnerabilidades. Transformações simples aplicadas a uma imagem, como deslocamentos por uma amostra, resultam em valores pobres de qualidade objetiva, apesar de os sinais permanecerem subjetivamente idênticos.

## 2.3 TÉCNICAS DE COMPRESSÃO DE VÍDEO

Uma vez disponíveis sinais digitais de vídeo, é possível fazer o uso de diversas técnicas de compressão a fim de representá-los convenientemente. A primeira idéia que vem à mente ao se pensar em comprimir sinais de vídeo é comprimir cada quadro da seqüência separadamente usando técnicas de compressão de imagens como o codificador JPEG [22]. Seu princípio básico de funcionamento consiste na fragmentação da imagem em pequenos blocos de tamanhos iguais; para cada bloco, aplica-se a transformada discreta de cosseno (DCT).

### 2.3.1 A Transformada Discreta de Cosseno (DCT): viabilização da compressão.

A DCT [23] é uma transformada muito usada em padrões de compressão de vídeo devido a sua capacidade de decorrelação e a eficiência na compactação de energia. Por capacidade de decorrelação entende-se a habilidade de uma transformada converter um conjunto de dados altamente correlacionados em outro conjunto de dados relativamente independentes (reduzindo a redundância estatística) enquanto eficiência em compactação de energia refere-se à habilidade de uma transformada compactar o conteúdo energético de um sinal na menor quantidade possível de coeficientes. O fato de a DCT ser uma transformada independente de dados e a disponibilidade de implementações rápidas são argumentos que justificam sua popularidade. [24]

Para um sinal de  $N \times N$  amostras, tomado aqui como uma fração (bloco) de um quadro a ser comprimido, os coeficientes de sua matriz de transformação  $\mathbf{C}$  são obtidos por funções de cossenos [4], conforme verificado na Equação 2.3:

$$[\mathbf{C}]_{ij} = \begin{cases} \sqrt{\frac{1}{N}} \cos \frac{(2j+i)i\pi}{2N} & i = 0; j = 0, 1, \dots, N-1 \\ \sqrt{\frac{2}{N}} \cos \frac{(2j+i)i\pi}{2N} & i = 1, 2, \dots, N-1; j = 0, 1, \dots, N-1. \end{cases} \quad (2.3)$$

O resultado da aplicação da DCT em um bloco de  $N \times N$  amostras é um conjunto de  $N \times N$  coeficientes representando o bloco no domínio transformado, coeficientes que podem ser considerados como ponderações para um conjunto de matrizes de base, ilustrado na Figura 2.1 para o caso em que  $N = 8$ . Dessa forma, a representação no domínio transformado pode ser encarada como a representação do sinal pela combinação de todas as  $N \times N$  matrizes de base, cada qual multiplicada por seu fator de ponderação apropriado. [2]

### 2.3.2 Quantização e Codificação de Entropia: realização da compressão.

Aos coeficientes transformados pela DCT é aplicado um processo de quantização, responsável pela perda seletiva de informações no processo de compressão. No término do processo, os coeficientes quantizados passam por um processo de codificação de entropia.

Apesar de bastante simples, o método descrito anteriormente ainda é encontrado em codificadores de vídeo considerados estado da arte em tecnologia e é comumente denominado de processo de codificação **Intra-quadro**, uma vez que as informações usadas na compressão do sinal não dependem de quadros anteriores. O próprio codificador o JPEG2000 [25], sucessor do JPEG e considerado estado da arte na

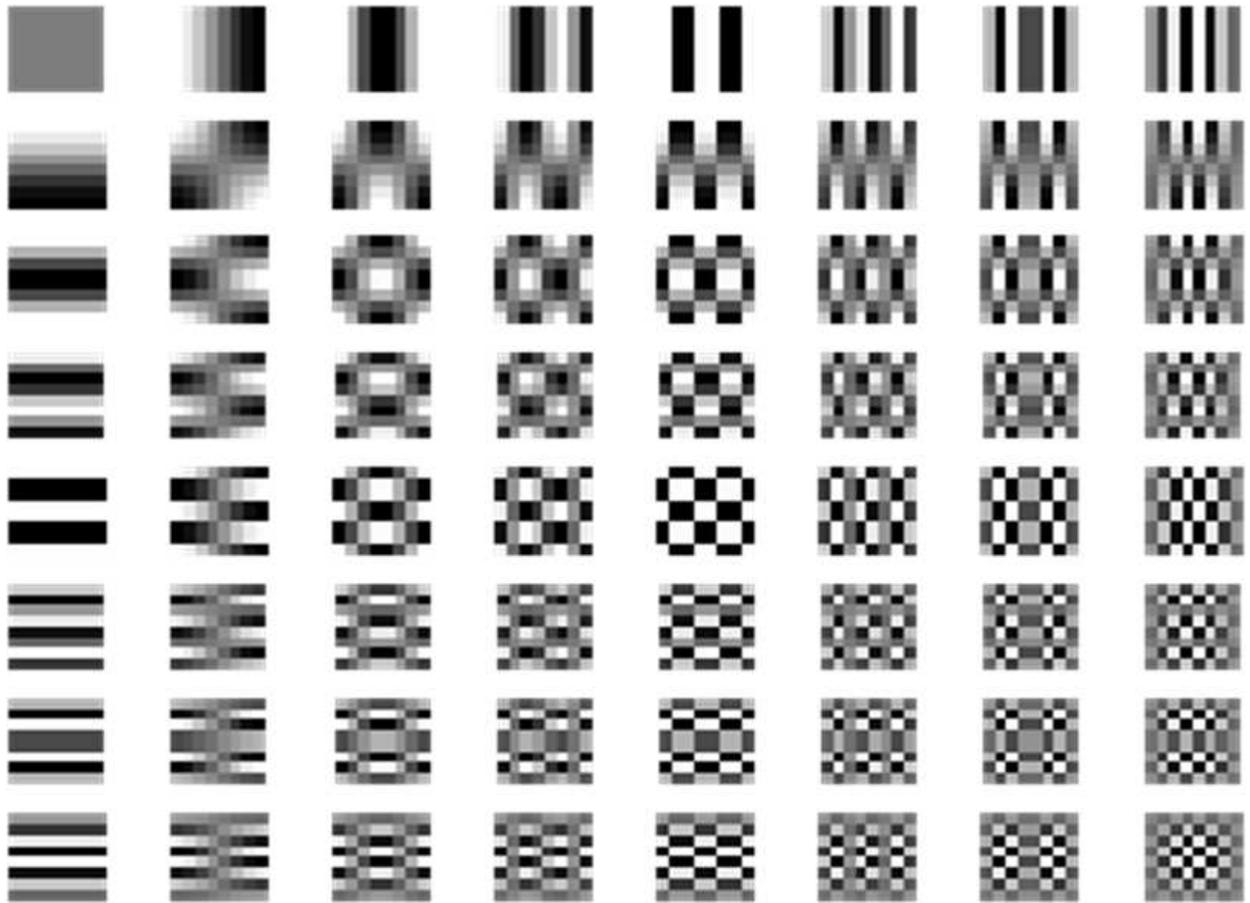


Figura 2.1: Bases da DCT para blocos de  $8 \times 8$  pixels. Adaptado de [1].

compressão de imagens estáticas, é tomado como alternativa na codificação de vídeo para aplicação de cinema digital [26]. Todavia, um grande desempenho em compressão ainda pode ser obtido se levarmos em consideração a redundância temporal em um sinal de vídeo.



Figura 2.2: Dois quadros consecutivos da seqüência Foreman.

Como podemos perceber na Figura 1.1, repetida aqui na Figura 2.2, muito do conteúdo visual da cena é repetido entre quadros consecutivos sem mudanças significativas. Uma maneira mais eficiente de representar o sinal de vídeo seria pela codificação somente das mudanças no conteúdo de vídeo em vez de codificar o quadro completamente. O conjunto de técnicas que exploram redundâncias temporais, usualmente denominado codificação **Inter-quadros**, é responsável pelo grande ganho de codificação que distingue um codificador de vídeo de um codificador de imagens aplicado seqüencialmente.

No processo de codificação Inter-quadros, usualmente se fragmenta o quadro em pequenos blocos de igual tamanho e verifica-se se há diferença entre o bloco do quadro atual e o bloco que ocupa a mesma posição no quadro anterior. Havendo diferenças, essa é calculada, passa por um processo de codificação muito semelhante ao de um bloco em codificação Intra-quadro e é enviada. Tal metodologia pode ser refinada se uma análise mais datalhada da correlação existente entre os quadros for realizada por meio de uma técnica de predição chamada **compensação de movimentos**.

A maioria das mudanças existentes entre os quadros de vídeo são tipicamente devidas ao movimento de objetos da cena em relação a um fundo estático. Pequenas quantidades de movimentos podem resultar em grandes diferenças entre blocos co-localizados em quadros consecutivos. Tipicamente, a busca de um bloco feita numa janela ao redor da posição do bloco no quadro anterior, chamado de quadro de referência, pode reduzir significativamente a quantidade de informação necessária para codificar adequadamente o quadro. O uso de deslocamentos espaciais como forma de aproximação do sinal é denominado **compensação de**

**movimentos** e ao processo de busca pelo melhor casamento entre blocos do quadro atual e dos anteriores dá-se o nome de **estimação de movimentos**. Nesse caso, além das diferenças, também chamadas de resíduo de predição, é necessário o envio informações que indiquem qual foi o bloco do quadro anterior usado na aproximação; a essa informação lateral de deslocamento dá-se o nome de **vetor de movimento**.

Na Figura 2.3 é apresentado um campo vetorial com os vetores de movimento que indicam de qual bloco (aqui de tamanho de  $8 \times 8$  *pixels* ou pontos elementares de imagem) do quadro anterior provém o melhor casamento com o bloco do quadro atual e na Figura 2.4 são comparadas a versão original do quadro atual e sua versão predita a partir de compensação de movimentos em relação ao quadro anterior. Observa-se na Figura 2.5 como a energia do resíduo entre o quadro e sua versão predita por compensação de movimentos é bem menor do que o resíduo entre o mesmo quadro e o subsequente. É esse resíduo, juntamente com os vetores de movimento, que precisa ser enviado ao decodificador para garantir a reconstrução do sinal de vídeo do outro lado do canal.

Em um codificador de vídeo, os parâmetros extraídos pelo processo de compressão (coeficientes quantizados, vetores de movimento etc.) precisam ser representados eficientemente e sem erros de forma que o decodificador possa recompô-los e decodificar o sinal tal qual foi reconstruído no codificador. Para comprimir essas informações sem perdas, usa-se codificação de entropia. [4] As técnicas de codificação entrópica apresentam bons resultados para fontes de informação sem memória (fontes cujo valor de cada amostra não apresentam dependência entre si) minimizando a taxa de bits necessária para a codificação pela associação de códigos de comprimento variável para amostras de entrada de acordo com a função de densidade de probabilidade (*fdp*) das amostras. O princípio básico de operação é simples: para as amostras mais recorrentes usam-se símbolos mais curtos enquanto às amostras menos freqüentes são associados símbolos mais longos em bits.

Assumir que fontes de informação não possuem memória é uma abordagem válida, porém simplista. A fim de extrair o máximo da redundância existente em dados a serem comprimidos, é possível conjugar a codificação entrópica a codificadores preditivos, que tratam mais adequadamente fontes com memória, ou seja, fontes em que cada amostra apresenta dependência estatística em relação às amostras adjacentes. Essa classe de codificadores reduz a entropia e é aplicada em larga escala em compressores com perdas.

### **2.3.3 Codificador de Vídeo DPCM Híbrido**

Ao agregado de técnicas de estimação/compensação de movimentos, de codificação de imagens e de entropia empregado em um codificador de vídeo dá-se o nome de codificação de vídeo híbrida. As Figuras

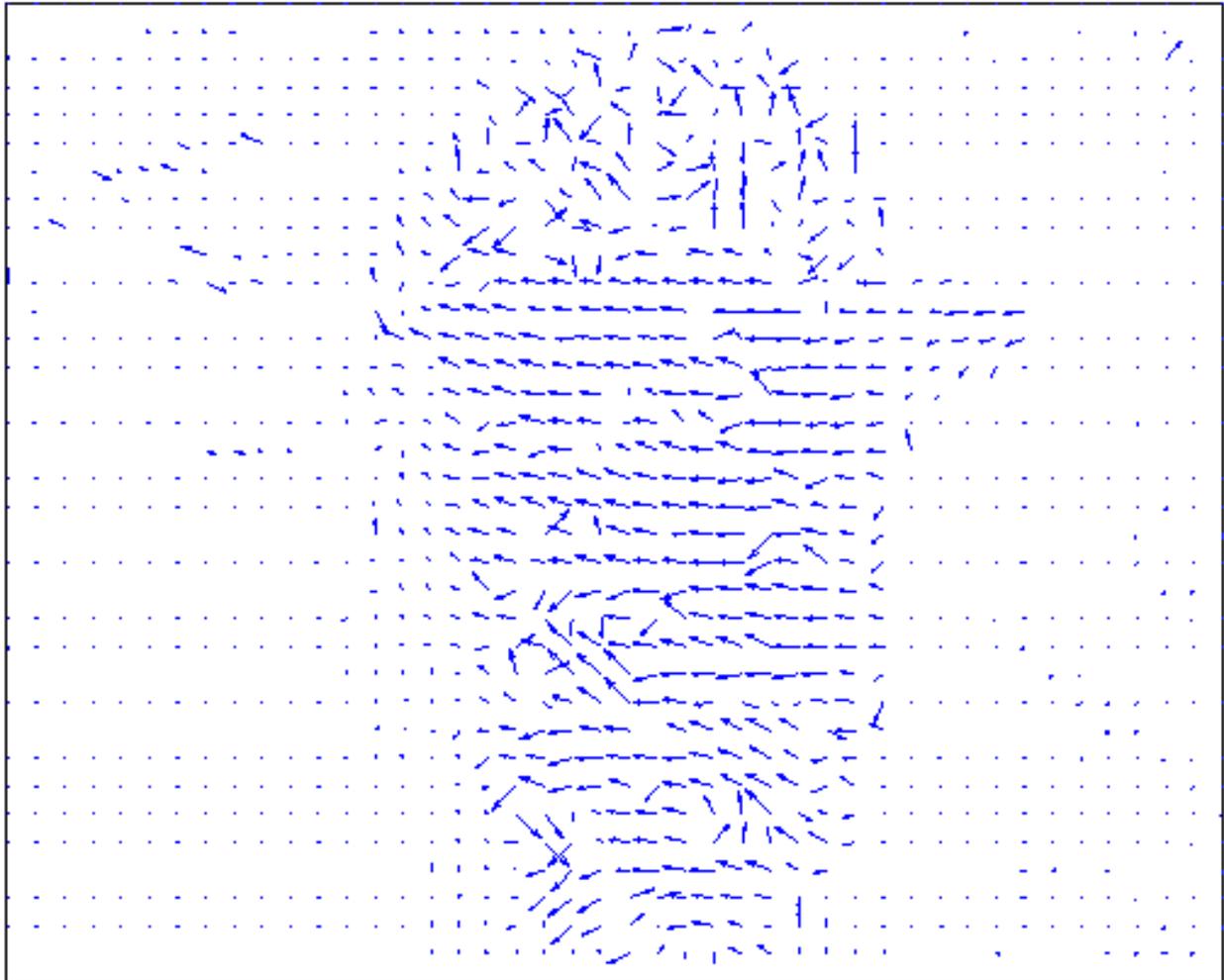


Figura 2.3: Fluxo óptico composto por vetores de movimento para cada bloco de  $8 \times 8$  *pixels* da imagem da Figura 2.2(b) com relação à Figura 2.2(a).



(a)

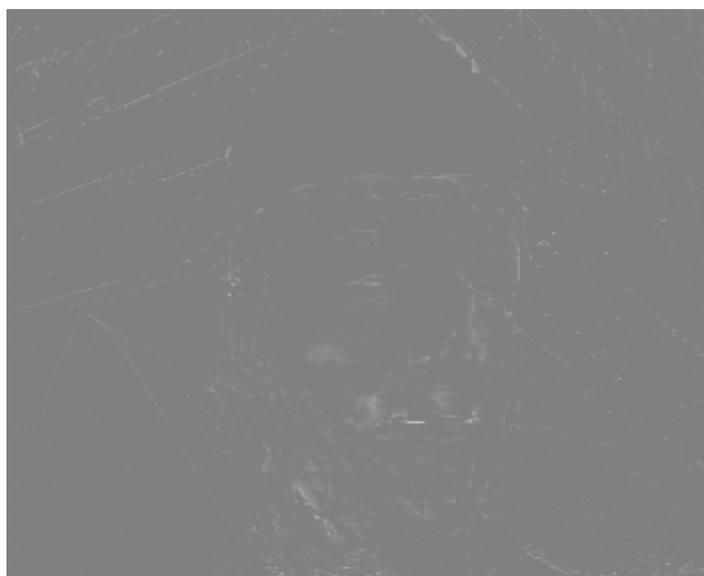


(b)

Figura 2.4: Versões do quadro da Figura 2.2(b): (a) original e (b) compensado com relação à Figura 2.2(a) empregando o fluxo óptico da Figura 2.3.



(a)



(b)

Figura 2.5: Resíduo entre os quadros consecutivos (a) sem predição e (b) utilizando predição por compensação de movimentos.

2.6 e 2.7 apresentam diagramas de um codificador e de um decodificador DPCM [4] híbrido baseado em DCT.

O codificador do tipo DPCM é uma estrutura muito usada no tratamento de fontes que apresentam um elevado grau de correlação entre as amostras. Ele procura explorar a correlação para prever novas amostras a partir de amostras já comprimidas e codificar e transmitir apenas as diferenças entre a predição e o novo valor. O decodificador DPCM deve “imitar” (seção 2.3.3.2) o codificador para gerar predições idênticas e adicioná-las às diferenças enviadas pelo codificador, resultando nas amostras decodificadas.

No codificador da Figura 2.6, um quadro de vídeo ( $Q_n$ ) é processado para produzir o trem de bits (*bitstream*) comprimido e, no decodificador (Figura 2.7), o *bitstream* comprimido é decodificado para produzir o quadro reconstruído ( $Q'_n$ ) ligeiramente diferente do quadro original. Nas figuras estão realçados elementos comuns ao codificador e decodificador.

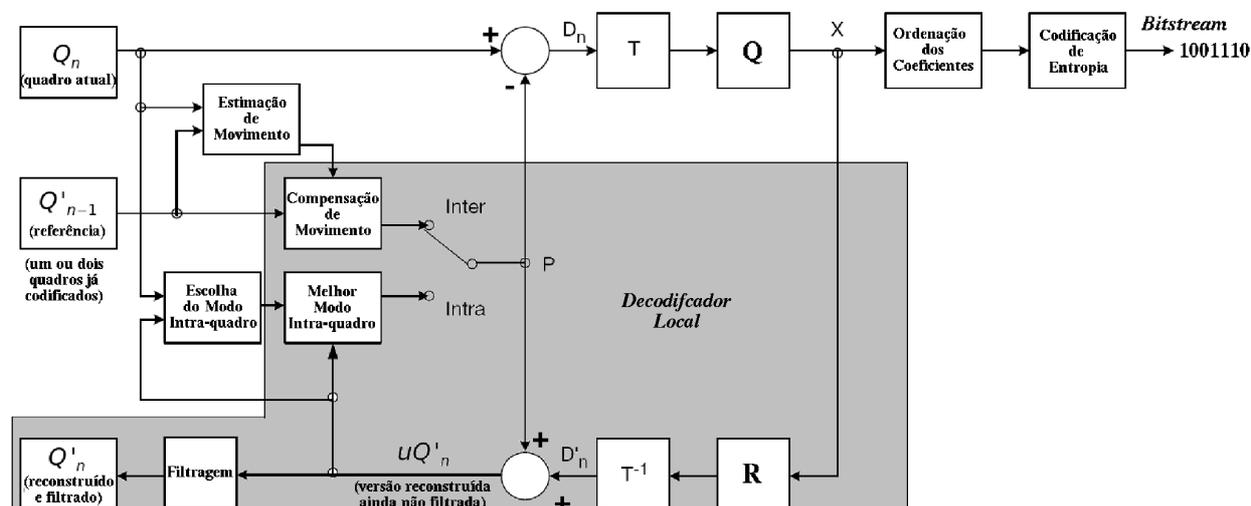


Figura 2.6: Codificador de vídeo DPCM híbrido. Adaptado de [2].

### 2.3.3.1 Fluxo de dados do codificador

Há dois fluxos principais de dados no codificador: o caminho de codificação e o caminho de decodificação. Os passos de codificação são os seguintes:

1. Um quadro de entrada  $Q_n$  entra no codificador e é processado em pequenas frações, chamadas macroblocos, tomados aqui como blocos com  $16 \times 16$  pixels.
2.  $Q_n$  é comparado com um quadro de referência que pode ser o quadro codificado anteriormente ( $Q'_{n-1}$ ). Uma função de estimação de movimentos encontra uma região de  $16 \times 16$  amostras em

$Q'_{n-1}$  que casa com o macrobloco de  $Q_n$  que está sendo codificado no momento. O deslocamento entre as posições do macrobloco atual e o de melhor casamento do quadro de referência é o vetor de movimento  $VM$ .

3. A partir do vetor de movimento, uma predição  $P$  por compensação de movimentos é gerada: predição Inter-quadros. É possível gerar o sinal  $P$  a partir de extrapolação espacial dos seus vizinhos contidos numa janela causal: predição Intra-quadro.
4. A melhor predição encontrada pelos processos Intra ou Inter-quadros é tomada ( $P$ ) e subtraída do macrobloco atual para produzir o macrobloco residual  $D_n$ .
5.  $D_n$  é partido em blocos menores e passa por um processo de transformação, por exemplo transformado pela DCT.
6. A versão transformada de  $D_n$  é quantizada ( $X$ ), ou seja, os coeficientes transformados passam por um mapeamento em que a intervalo de definição de seus valores é reduzido para um conjunto finito menor e que requer menos bits para ser representado. É nesta etapa em que ocorrem as perdas irreversíveis do processo de codificação.
7. Os coeficientes de cada bloco são rearranjados e submetidos à codificação de entropia juntamente com informações a respeito do tipo de predição empregado, seus parâmetros (vetores de movimento ou modo de predição espacial) e informações de cabeçalho, produzindo o *bitstream* comprimido.

Os passos de reconstrução são:

1. Cada macrobloco quantizado  $X$  é re-escalonado (R), processo em que os valores de seus coeficientes transformados são mapeados de volta para o intervalo de definição original. Em seguida, é submetido a transformada inversa da DCT para obtenção do resíduo decodificado  $D'_n$ . Note que o processo de quantização não é reversível, o que significa que  $D'_n$  não seja idêntico a  $D_n$ . A diferença entre essas duas versões é a causa da **distorção** no processo.
2. Independentemente do tipo de predição empregado, Intra-quadro ou Inter-quadros, o sinal predito  $P$  é adicionado ao resíduo  $D'_n$  para produzir um macrobloco reconstruído e os macroblocos reconstruídos são salvos para a montagem do quadro reconstruído  $Q'_n$ .
3. Antes de ser exibido e guardado na memória de quadros de referência, o sinal ainda passa por um processo de filtragem para a obtenção do quadro filtrado  $Q'_n$ .

Uma vez terminada a codificação completa do quadro, o quadro  $Q'_n$  pode ser usado como quadro de referência para o próximo quadro da seqüência de vídeo,  $Q'_{n+1}$ . Note que esta estrutura é muito semelhante à do decodificador, o que justifica sua comum denominação de decodificador local.

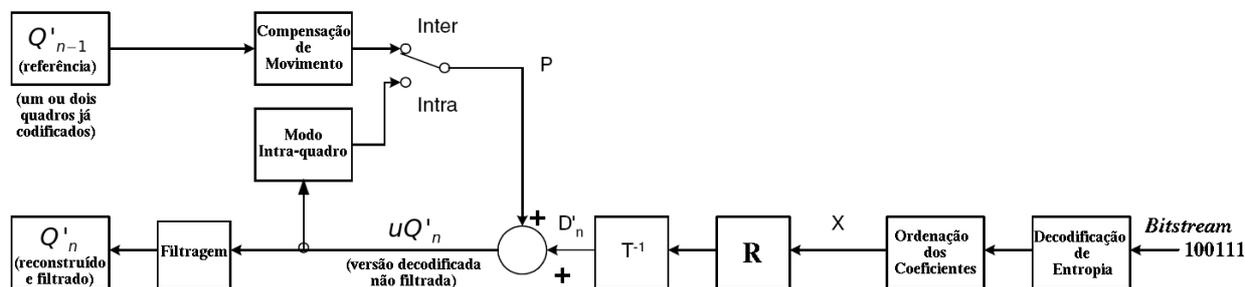


Figura 2.7: Decodificador de vídeo DPCM híbrido. Adaptado de [2].

### 2.3.3.2 Fluxo de dados do decodificador

Por sua vez, no decodificador há apenas um fluxo de dados, descrito a seguir:

1. O *bitstream* comprimido é decodificado entropicamente para a extração dos coeficientes, parâmetros de predição (vetores de movimento ou direção de predição espacial) e cabeçalhos de cada macrobloco. Os coeficientes são rearranjados para produzir a versão quantizada do macrobloco,  $X$ .
2.  $X$  é re-escalado e a ele é aplicada a transformada inversa para gerar o resíduo decodificado  $D'_n$ .
3. O vetor de movimento decodificado é usado para localizar a região de  $16 \times 16$  amostras na cópia do quadro de referência  $Q'_{n-1}$  existente no decodificador. Essa região é a predição por compensação de movimentos  $P$  para o caso Inter-quadros. Caso a predição sinalizada seja Intra-quadro, é realizada a composição do sinal  $P$  pela extrapolação espacial a partir de vizinhos em uma janela causal de codificação.
4. O sinal predito  $P$  é adicionado ao resíduo  $R'$  para produzir o macrobloco reconstruído e os macroblocos reconstruídos são armazenados para a montagem do quadro reconstruído  $Q'_n$ .

Depois que o quadro for completamente decodificado,  $Q'_n$  está pronto para exibição e pode ser armazenado como quadro de referência para o próximo quadro a ser decodificado,  $Q'_{n+1}$ .

## 2.4 O CODIFICADOR H.264/AVC

H.264/AVC é o estado da arte em codificação de vídeo. Ele foi desenvolvido em trabalho conjunto entre *Video Coding Experts Group* (VCEG, ITU-T) e *Moving Picture Experts Group* (MPEG, ISO/IEC) [12]. Esse padrão apresenta uma série de melhoramentos [27] em relação a padrões anteriores que resultam em ganhos de codificação para uma vasta gama de aplicações, entre elas a vídeo conferência, TV digital, cinema digital, difusão de vídeo etc. A seguir serão apresentados os refinamentos em relação à estrutura de codificador híbrido agregados ao H.264/AVC.

### 2.4.1 Predição Intra-quadro

A predição Intra-quadro é a técnica pela qual amostras do sinal de vídeo agrupadas em macroblocos de  $16 \times 16$  *pixels*, tomados no H.264/AVC como unidade básica de codificação, são preditos usando informações de macroblocos já codificados e pertencentes ao mesmo quadro. Nos perfis mais básicos do H.264/AVC (vide seção 2.4.6) existem inicialmente dois tipos de predição Intra-quadro para as componentes de *Y* luminância: INTRA\_4×4 e INTRA\_16×16. Com o advento do adendo FRExt (*Fidelity Range Extensions*) [28], foi acrescentado mais um tipo de predição, INTRA\_8×8, para o perfil *High*.

Os tipos Intra\_4×4 e Intra\_8×8 particionam o macrobloco em blocos menores de tamanhos 4×4 ou 8×8 *pixels*, respectivamente. Para cada um desses blocos, são disponibilizados nove modos de predição em que técnicas espaciais são empregadas. Um modo de predição é o DC em que todas as amostras do sub-bloco de 4×4 são preditas pela média das amostras vizinhas à esquerda (A) ou acima (B, C e D) do bloco atual (veja Figura 2.8(a)) e que já tenham sido reconstruídas no codificador e no decodificador. Os outros oito modos exploram direções para predição planar, mostradas na Figura 2.8(b), também a partir da referida janela causal.

No tipo INTRA\_16×16, o macrobloco completo é predito por apenas um modo. Na Figura 2.8(c) são apresentados os quatro modos suportados: predição vertical, predição horizontal, predição DC e predição planar. A predição Intra-quadro por macrobloco completo é muito eficiente quando o sinal a ser codificado apresenta variações suaves.

Quando um quadro é inteiramente codificado por meio do uso de técnicas de predição Intra-quadro, ele é chamado de quadro I.

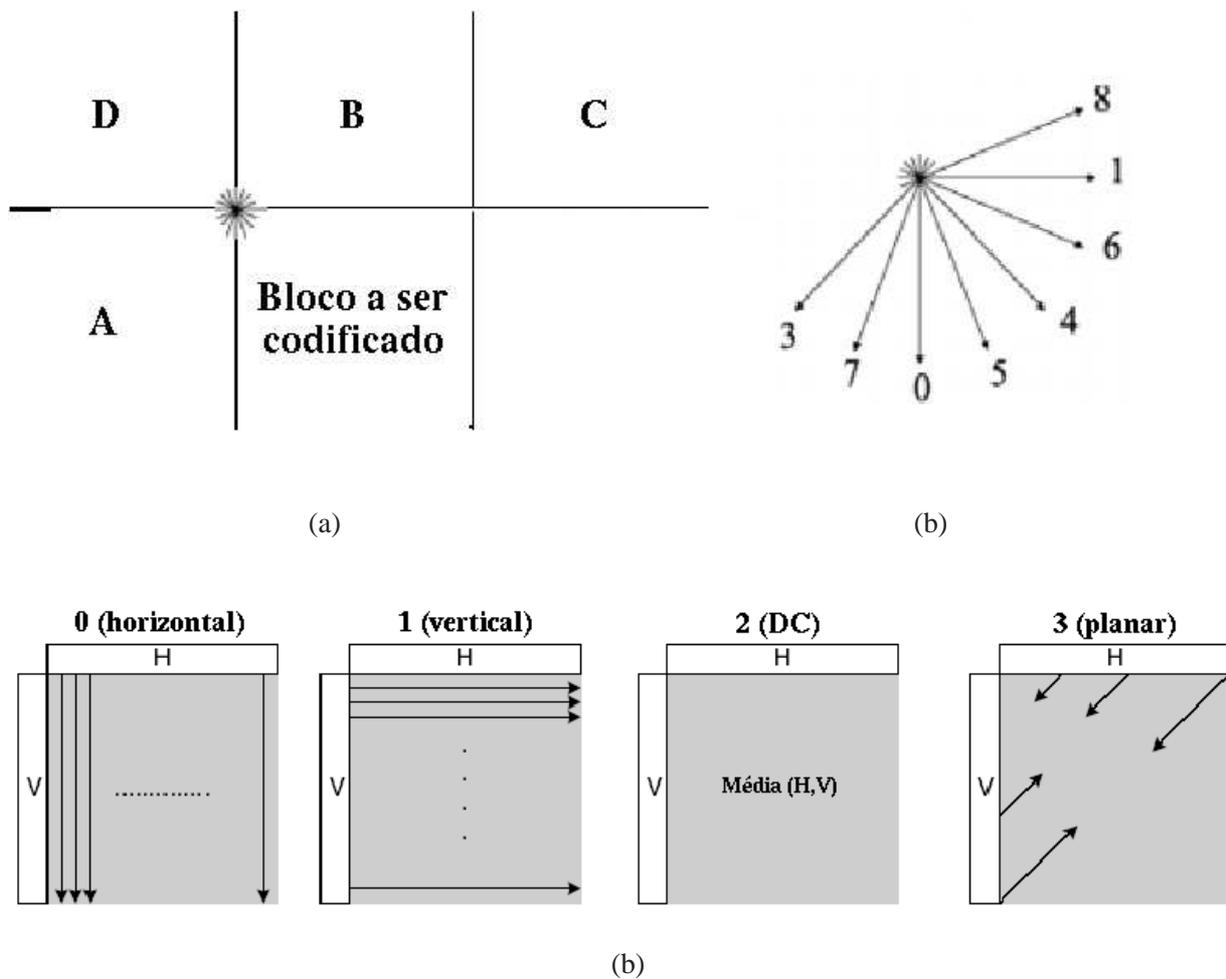


Figura 2.8: Janela causal (a) para codificação e modos de predição para codificação Intra-quadro com suas direções de interpolação planar para blocos  $4 \times 4$  e  $8 \times 8$  (b) e  $16 \times 16$  (c). O modo 2, não apresentado em (b), é o DC.

## 2.4.2 Predição Inter-quadros

Quando é mais interessante lançar mão de informações de quadros anteriores na geração da predição de um macrobloco, a compensação de movimentos é empregada como técnica para prever o sinal a partir de quadros já transmitidos, denominados quadros de referência. No H.264/AVC quadros que, além de usar técnicas de predição espacial, empregam técnicas de predição temporal são classificados em duas categorias: quadros P e quadros B.

Um dos grandes avanços em relação a outros padrões vem do fato de que cada macrobloco pode ser dividido em blocos menores, chamados de partições. Os tamanhos possíveis de partição são:  $16 \times 16$ ,  $16 \times 8$ ,  $8 \times 16$  e  $8 \times 8$  *pixels*; a partição de  $8 \times 8$  pode ainda ser subdividida em partições menores de  $8 \times 4$ ,  $4 \times 8$  e  $4 \times 4$  *pixels*. As partições são ilustradas na Figura 2.9.

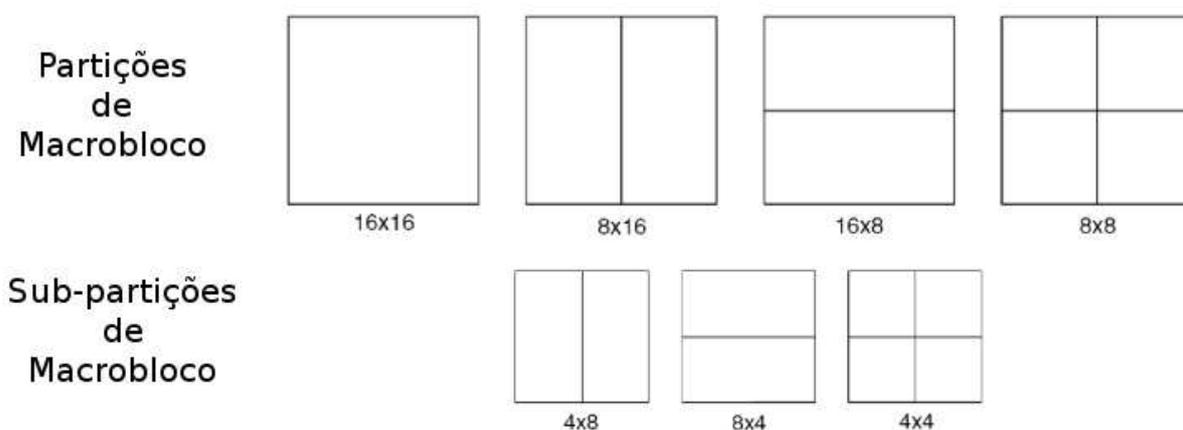


Figura 2.9: Partições de macroblocos.

A maneira como foram propostas as partições de macrobloco da Figura 2.9 permite estimação de movimento refinada pois movimentos de objetos complexos ficam melhor mapeados pela maior variedade de tamanhos e formatos dos blocos. Além disso, os tamanhos utilizados fazem conveniente representação hierárquica em uma estrutura denominada *quadtree*, apresentada na Figura 2.10.

A precisão no cálculo dos vetores de movimento é outro aspecto bem explorado no H.264/AVC, em que os vetores de movimento são calculados com precisão de até um quarto de *pixel* ( $1/4$ -*pel*). Vetores de movimento com resolução fracionária referem-se a posições dos quadros de referência que estão situadas entre as amostras do quadro digital. Para gerar vetores de movimento fracionários empregados na compensação, o sinal do quadro de referência precisa ter preenchidas as posições sub-*pixel* por meio de interpolação.

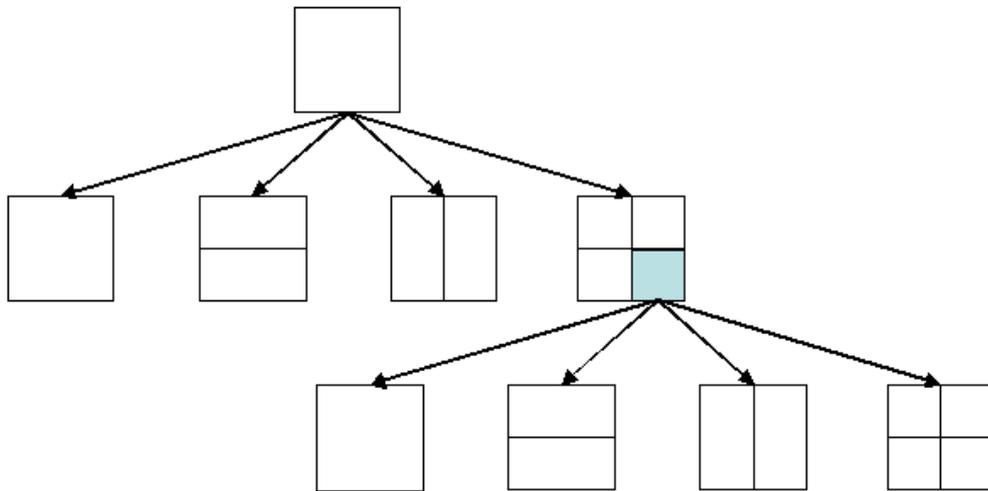


Figura 2.10: Organização das partições em uma estrutura hierárquica. Adaptado de [1].

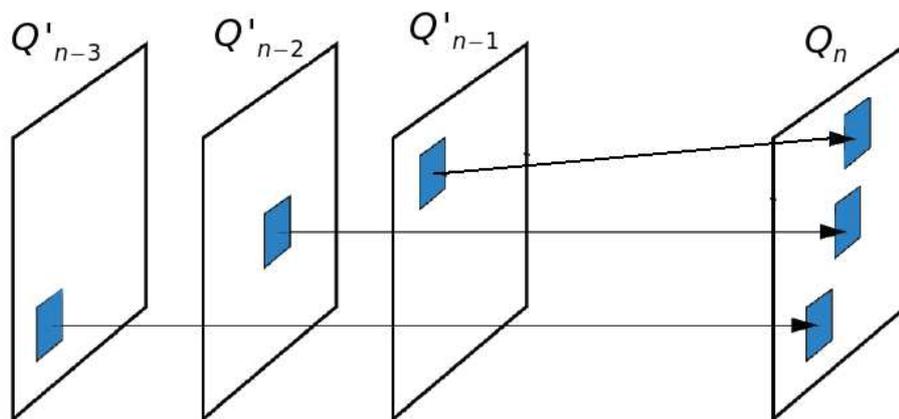


Figura 2.11: Compensação de movimento com múltiplos quadros de referência em quadros P.

Padrões anteriores permitiam apenas um número limitado de partições, por exemplo,  $16 \times 16$  e  $8 \times 8$  pixels no caso do H.263[9] ou MPEG-4[10]. Vetores deslocamento são estimados e transmitidos para cada um desses blocos e referem-se à posição da região de melhor casamento com o bloco atual em um quadro de referência, usualmente o último quadro reconstruído. No H.264/AVC é possível fazer referência a várias imagens precedentes, sendo necessário o envio de informações dos quadros de referência empregados juntamente com os vetores de movimento ao decodificador. A esta técnica dá-se o nome de previsão por compensação de movimento com múltiplos quadros de referência, a qual é ilustrada nas Figuras 2.11 e 2.12.

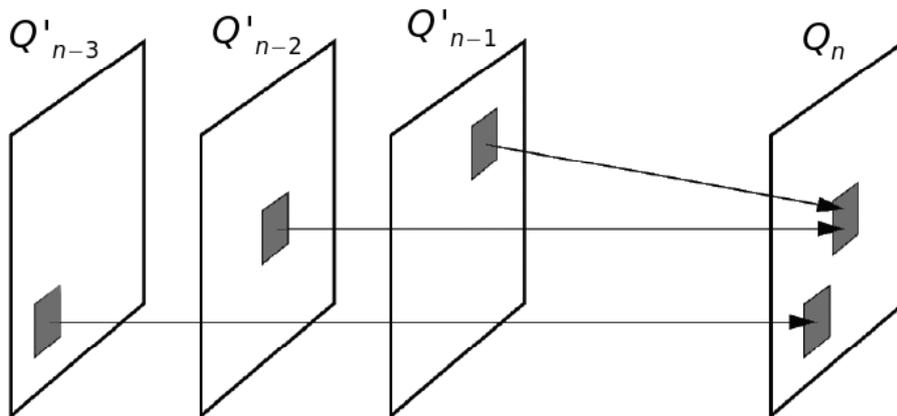


Figura 2.12: Compensação de movimento com múltiplos quadros de referência em quadros B.

Chamamos quadros B ou *B-slices* os quadros codificados com macroblocos usando múltiplas referências simultaneamente (Figura 2.12). Comparado a padrões de compressão de vídeo anteriores, o conceito clássico de quadros bi-preditos é generalizado no H.264/AVC ao se permitir uma combinação linear de duas previsões por meio da aplicação de pesos arbitrários e deixar em aberto a direção das previsões temporais. Com isso, é possível explorar melhor as correlações temporais entre os quadros rearranjando-os de maneira conveniente e possibilitando que a ordem de codificação dos quadros seja diferente da ordem de exibição.

O quadro 4 da Figura 2.13 é um quadro P que emprega compensação de movimentos em sua codificação e usa como quadro de referência o quadro 1, um já codificado quadro I. Enquanto isso, os quadros 2 e 3, do tipo B, usam os quadros 1 e 4 simultaneamente para geração de suas previsões. Note que os quadros 2 e 3, por dependerem de versões codificadas dos quadros 1 e 4, são codificados posteriormente ao quadro 4, ilustrando o fato de a ordem de codificação não ser necessariamente ditada pela ordem de exibição. A ordem de codificação para o caso da Figura 2.13 é apresentada na Figura 2.14.

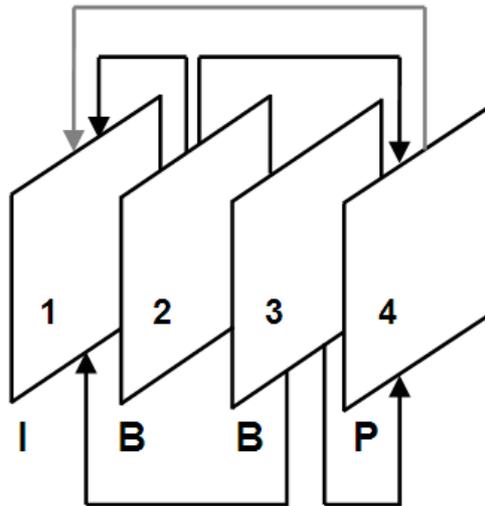


Figura 2.13: Quadros de compressão de vídeo ordenado de acordo com a seqüência de exibição. Adaptado de [1].

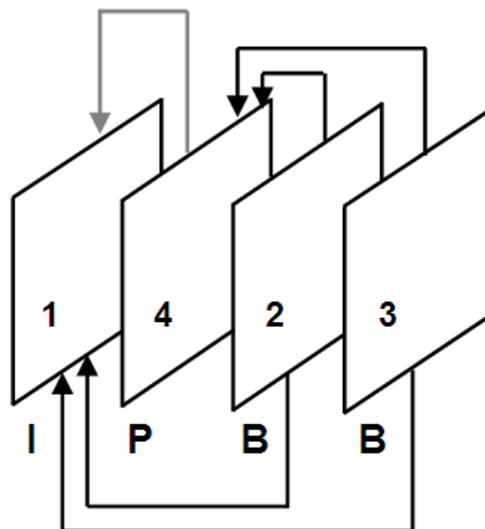


Figura 2.14: Quadros de compressão de vídeo ordenado de acordo com a seqüência de compressão. Adaptado de [1].

### 2.4.3 Codificação por transformadas

Seguindo a tendência dos padrões de compressão anteriores, a codificação por transformadas ainda é empregada para viabilizar a codificação do sinal residual por meio da redução da redundância espacial existente no erro de predição. Em vez de aplicar a DCT bidimensional em blocos de  $8 \times 8$  *pixels*, lugar comum para padrões como MPEG-1[6] e MPEG-2[7], no H.264/AVC são usadas transformadas inteiras reversíveis com tamanhos variados de  $8 \times 8$ ,  $4 \times 4$  a  $2 \times 2$  amostras. A possibilidade do emprego de transformadas menores que a DCT de  $8 \times 8$  permite ao compressor codificar melhor o resíduo de predição no caso em que são selecionadas partições de tamanhos menores que  $8 \times 8$ , o que ocorre freqüentemente na estimação de movimento de bordas de objetos, além de conseguir compactar melhor o sinal devido à menor variação encontrada em pequenas regiões de um quadro digital.

Levando em consideração apenas amostras de luminância, três tipos de transformadas são usados. O primeiro tipo é aplicado a todas as amostras do resíduo de predição, não importa se resultantes de predição por compensação de movimentos ou de predição Intra-quadro, e tem suporte  $4 \times 4$ . Se o macrobloco for predito usando INTRA\_16x16, é necessária a aplicação da segunda transformada de suporte  $4 \times 4$ , do tipo Hadamard [29], aos 16 coeficientes DC dos blocos de luminância já codificados [12]. Com o adendo FRExt[28], foi adicionada uma transformada inteira de  $8 \times 8$  para a predição Intra-quadro.

Os coeficientes transformados são quantizados escalarmente com passo de quantização determinado pelo parâmetro de quantização, QP, que pode assumir valores inteiros entre 0 e 51. Devido ao emprego de escala logarítmica, o passo de quantização duplica a cada incremento de 6 no QP, e um incremento unitário no QP resulta na redução de aproximadamente 12,5% na taxa de transmissão [13].

### 2.4.4 Codificação de Entropia

Uma das grandes vulnerabilidades das técnicas tradicionais de codificação de entropia vem da hipótese de que a estatística dos sinais seria estacionária e que seria possível levantar, de antemão, códigos de comprimento variável de tamanho ótimo para os elementos sintáticos normatizados. Essa hipótese raramente era verificada e nem sempre este tipo de compressão conseguia remover com eficiência a redundância do sinal. A incorporação de um modelamento de contextos no sistema de codificação de entropia é o que proporciona um alto grau de adaptação do H.264/AVC às fontes que serão comprimidas e contribui para o desempenho superior do padrão.

São dois os métodos de codificação de entropia disponíveis no H.264/AVC: uma técnica de baixa

complexidade baseada no uso de conjuntos de códigos de comprimento variável e de contexto adaptativo denominado CAVLC (*Context Adaptive Variable Length Codes*) e um algoritmo mais complexo que emprega codificação aritmética binária e de contexto adaptativo, CABAC (*Context Adaptive Binary Arithmetic Coding*)[30].

#### **2.4.5 Filtro de *Deblocking* Adaptativo**

A estrutura baseada no fracionamento de quadros em macroblocos, que, por sua vez, podem ser divididos em partições menores para compensação de movimentos e para aplicação de transformadas de blocos, sujeita o sinal de vídeo codificado a artefatos de bloco. A aplicação de filtros nas bordas dos blocos é tida como uma poderosa ferramenta para reduzir a percepção desses artefatos.

A filtragem para redução de efeitos de blocos, também chamada de filtragem de *deblocking*, era tomada como uma etapa posterior ao processo de codificação nos padrões MPEG-1, MPEG-2 e H.263, alterando o sinal apenas antes da apresentação do mesmo. Entretanto, uma qualidade visual superior pode ser alcançada por meio da aplicação da etapa de filtragem no laço de codificação o que implica que todos os quadros de referência sejam versões filtradas dos quadros reconstruídos, abordagem introduzida pelo H.261[5] e seguida pelo H.264/AVC[31] e pelo anexo J do H.263+ [32].

#### **2.4.6 Perfis do H.264/AVC**

O H.264/AVC foi desenvolvido para ser o mais genérico possível e capaz de atender uma vasta gama de aplicações, taxas, resoluções, qualidades e serviços. Todavia, diferentes aplicações apresentam demandas diferentes. De forma a maximizar a inter-operabilidade e proporcionar grande aceitação do padrão, a especificação do H.264/AVC define perfis e níveis.

Um perfil é definido como um subconjunto da sintaxe completa do *bitstream* ou, em outras palavras, um subconjunto de ferramentas de codificação. De forma a operar em um subconjunto da sintaxe, elementos sintáticos específicos para sinalização indicam a presença ou ausência de elementos sintáticos que venham a ser usados posteriormente no *bitstream*. Todos os decodificadores compatíveis com determinado perfil devem suportar as ferramentas de codificação correspondentes àquele.

Na primeira versão do H.264 três perfis foram definidos: *Baseline*, *Extended* e *Main*. Onde o perfil *Baseline* suporta todas as características do H.264/MPEG4-AVC, versão 1 (2003), exceto pelos conjuntos de características a seguir:

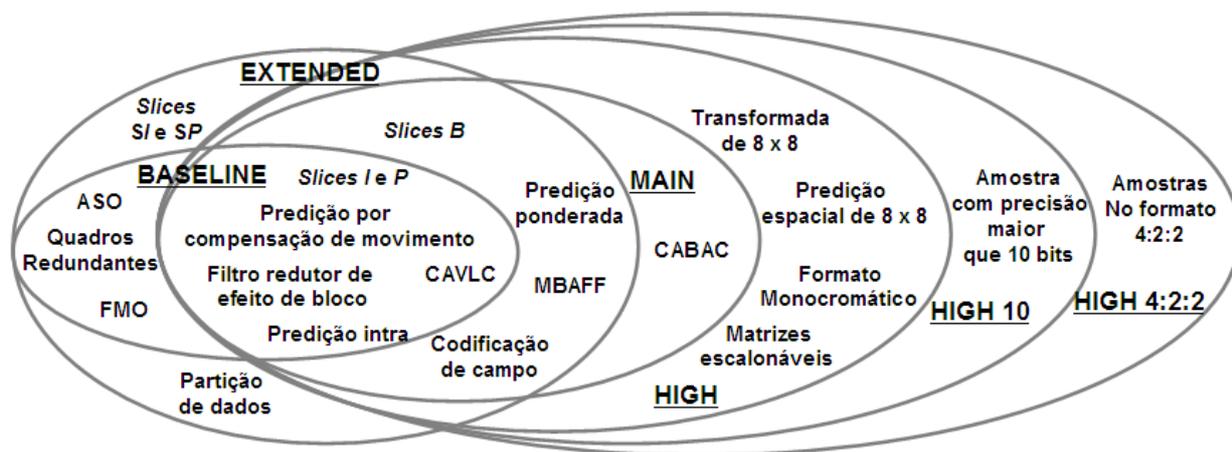


Figura 2.15: Ilustração dos perfis do H.264. Adaptado de [1].

1. *Slices B*, codificação de campo, chaveamento adaptativo entre quadro e campo (MBAFF - *Macroblock Adaptive Switching Between Frame and Field*) e predição ponderada.
2. CABAC.
3. *Slices SI (Switching I)* e *SP (Switching P)*, e partição de dados com *slice*.

Os dois primeiros itens contêm um conjunto de características que são suportados pelo perfil *Main*, em adição às características suportadas pelo *Baseline* exceto para a FMO (*Flexible Macroblock Order*) e outras características de robustez a erros [33]. E o perfil *Extended* suporta todas as características do perfil *Baseline* adicionado aos itens um e três. A grosso modo, o perfil *Baseline* foi desenvolvido visando aplicações com o mínimo de complexidade computacional e o máximo de robustez a erro, já o perfil *Main* focava aplicações que necessitassem do máximo em eficiência de compressão sem levar em consideração a complexidade computacional. Finalmente, o perfil *Extended* foi desenvolvido para promover um compromisso entre os perfis *Baseline* e *Main* com um foco para necessidades específicas de aplicações com *streaming* de vídeo adicionado à robustez e erros e perda de pacotes.

O padrão H.264/AVC sofreu um adendo, denominado FRExt [28], em que se adicionaram outros três perfis com base no perfil *Main*: *High*, *High 10* e *High 4:2:2*. Esses perfis foram desenvolvidos de forma a munir o codificador H.264/AVC de ferramentas demandadas por ambientes profissionais de edição de vídeo e com foco em resoluções mais elevadas. O perfil *High* possibilita o uso de sinais de vídeo de altas resoluções porém ainda não possibilita o emprego de formatos de cromaância mais precisos ou de amostras com maior acurácia; essa passa a ser disponibilizada pelo perfil *High 10* onde 10 bits são gastos por amostra do sinal. Os perfis *High 4:2:2* e *High 4:4:4* (previsto mas removido com a emenda [34],

por isso não ilustrado na Figura 2.15) representam cores com maior precisão permitindo amostragem de crominância do tipo 4:2:2 e 4:4:4, respectivamente.



# 3 AVALIAÇÃO DE COMPLEXIDADE DO CODIFICADOR

## H.264/AVC

### 3.1 INTRODUÇÃO

A maneira como foi proposto o estágio preditivo do H.264/AVC o fez ser oneroso computacionalmente pois inúmeros são os testes aplicados na escolha do melhor dos **modos de predição** disponíveis em termos de taxa  $\times$  distorção. É natural que uma redução do tempo de codificação de seqüências de vídeo digital deva passar pela modificação do estágio preditivo, especialmente pela etapa de estimação de movimentos.

Na literatura encontramos contribuições com enfoque no estágio de predição como estratégia de redução da complexidade computacional do codificador. Alguns trabalhos [35, 36, 37] resultaram em técnicas de estimação de movimento sub-ótimas que foram incorporadas ao código de referência do H264/AVC. A idéia básica foi empregar heurísticas para reduzir a quantidade de visitas necessárias durante a varredura da janela de busca na estimação de movimentos, propiciando ganhos consideráveis em tempo de codificação com pequenas perdas de qualidade do sinal em relação à técnica de busca completa.

Ainda atuando na estimação de movimentos, há propostas para redução de complexidade em que a estimação de movimentos extensiva é evitada e aplicada somente na partição de macrobloco mais provável, partição esta determinada por meio de heurísticas que exploram informações dos macroblocos previamente codificados. Em trabalho relacionado a esse tópico [38], a verossimilhança e a correlação dos campos de movimentos são aplicadas na seleção da melhor partição do macrobloco. Em outro trabalho [39] são empregadas informações sobre as partições de macroblocos de quadros anteriores para a formação de contexto usado na indicação das melhores partições, somente sobre as quais será aplicada estimação de movimentos.

Apesar de não representar parcela significativa da complexidade do codificador, os testes empregados para predição Intra-quadro podem ser reduzidos por meio da seleção prévia do melhor modo de acordo com heurísticas. [40, 41]

Outra linha encara a complexidade computacional como um recurso escasso e estende o conceito de otimização taxa *versus* distorção com a adição de um terceiro eixo na análise: o eixo da complexidade. Uma aplicação para esse tipo de abordagem é a comunicação de vídeo em sistemas sem fio com restrições

de energia. [15] Há ainda um foco na transmissão em tempo real de sinais de vídeo [42] desenvolvendo controle conjunto do preenchimento dos *buffers* de complexidade e de transmissão.

Observa-se que os trabalhos citados concentram-se em evitar testes exaustivos pela redução das possibilidades de modos de codificação para um subconjunto ótimo e, com isso, reduzir o tempo de compressão e a complexidade computacional.

### 3.2 PERFIL DE COMPLEXIDADE DO CODIFICADOR H.264/AVC HIGH PROFILE

A fim de verificarmos o perfil de complexidade do codificador H.264/AVC *High Profile* e confirmarmos as estimativas de complexidade apresentadas em [14], usou-se de ferramentas de programação chamadas de *profiler*, empregadas no levantamento de informações a respeito de um programa tais como: funções usadas numa tarefa, o tempo gasto por cada uma delas, a quantidade de vezes em que foram chamadas, suas interrelações e quantidade de memória usada.

Para o sistema operacional *Linux* há a ferramenta `gprof`, um *profiler* que se enquadra na categoria de *software livre* e é capaz de indicar o tempo gasto em cada função de um programa. Ele se fez útil no levantamento do perfil do H.264/AVC *High Profile* para a compressão da seqüência de vídeo HD Pedestrian Area com os seguintes parâmetros: quadros de  $1920 \times 1080$  *pixels*, otimização taxa  $\times$  distorção habilitada e *buffer* de referência com 4 quadros. Os resultados obtidos estão sumarizados nas Tabelas 3.1 e 3.2.

Tabela 3.1: Complexidade computacional relativa para codificação somente Intra-quadro da seqüência HD Pedestrian Area pelo codificador H.264/AVC *High Profile* com otimização taxa  $\times$  distorção ativa.

Etapa de Codificação	Percentual [%]
Predição Intra $4 \times 4$	25,2
Predição Intra $8 \times 8$	20,9
Predição Intra $16 \times 16$	6,1
Demais Etapas	47,8
Total	100,0

A Tabela 3.1 apresenta as contribuições de complexidade dos métodos de predição Intra-quadro disponíveis no codificador H.264/AVC. O que se verifica é que a complexidade da predição Intra-quadro para blocos de tamanhos  $4 \times 4$  e  $8 \times 8$  *pixels* são muito parecidas entre si e bem maiores que a predição para

Tabela 3.2: Complexidade computacional relativa para codificação da seqüência HD Pedestrian Area pelo codificador H.264/AVC *High Profile* com otimização taxa  $\times$  distorção ativa e variados tamanhos de janela de busca para estimação de movimentos.

Etapa de Codificação	Janela				
	8	16	32	64	128
Estimação de Movimentos	65,8%	78,5%	90,8%	97,1%	99,1%
Predição Intra 4 $\times$ 4	6,1%	3,8%	1,6%	0,5%	0,1%
Predição Intra 8 $\times$ 8	4,9%	3,2%	1,3%	0,4%	0,1%
Predição Intra 16 $\times$ 16	1,4%	1,0%	0,4%	0,1%	0,0%
Demais Etapas	21,8%	13,5%	5,9%	1,9%	0,7%
Total	100,0%				

blocos de 16 $\times$ 16 *pixels*. Os resultados confirmam o que é apresentado na definição do padrão H.264/AVC [12]: são disponibilizados quatro modos de predição Intra-quadro para macroblocos de tamanho 16 $\times$ 16 *pixels* e nove modos de predição Intra-quadro para macroblocos de tamanho 4 $\times$ 4 e 8 $\times$ 8 *pixels*.

Os resultados de complexidade do trabalho [14] puderam ser confirmados na Tabela 3.2 em que diferentes tamanhos de janela são aplicados para codificação de uma seqüência HD com o codificador H.264/AVC *High Profile*, que gasta a maioria do tempo de codificação no módulo de predição, com a etapa de Estimação de Movimentos dominando a complexidade computacional usada não só pelo módulo de predição, mas também por todo o codificador.

### 3.2.1 Predição com o Laço Aberto

A observação das Tabelas 3.1 e 3.2 mostra que o H.264/AVC gasta a maioria do tempo de codificação no módulo de predição, particularmente na etapa de Estimação de Movimentos. No presente trabalho, propõe-se a redução no tempo de codificação por meio da paralelização do codificador H.264/AVC, uma estrutura notadamente seqüencial. Para tanto, propõe-se modificar o estágio de predição de maneira que sua etapa mais complexa, a estimação de movimentos na codificação Inter-quadros, seja particionada em sub-módulos executados simultaneamente por várias máquinas de codificação. Para tanto, nesta dissertação propõe-se abrir o laço de codificação conforme descrito na Seção 4.1.

### 3.3 POLARIZAÇÃO DOS MODOS DE PREDIÇÃO

O codificador de vídeo H.264/AVC é estruturado como um codificador DPCM híbrido. Como mostrado em [14] e confirmado na Seção 3.2, o módulo de predição é o mais complexo pois nele são testados os vários modos de predição disponíveis para um macrobloco. É às custas de complexidade computacional que o codificador H.264/AVC é bem sucedido na representação comprimida dos sinais de vídeo.

No tratamento de vídeo de alta resolução, em que os quadros de uma seqüência são compostos por 1920 *pixels* de largura por 1080 *pixels* de altura, é comum a polarização dos modos de predição em classes bem determinadas. Tomemos o caso da codificação de quadros P em que as técnicas Inter-quadros são combinadas com as técnicas Intra-quadros e cujo conjunto de modos de predição disponíveis é composto por:

- P16x16: predição por compensação de movimentos para macroblocos de 16x16 *pixels*;
- P16x8: predição por compensação de movimentos para macroblocos de 16x8 *pixels*;
- P8x16: predição por compensação de movimentos para macroblocos de 8x16 *pixels*;
- P<=8x8: predição por compensação de movimentos para macroblocos menores ou de tamanho 8x8 *pixels*;
- I16MB: predição Intra-quadro para macroblocos de 16x16 *pixels*;
- I8MB: predição Intra-quadro para macroblocos de 8x8 *pixels*;
- I4MB: predição Intra-quadro movimentos para macroblocos de 4x4 *pixels*;
- SKIP: tipo de predição em que o bloco é de 16x16 *pixels*, o modo de predição é ditado pelos modos vizinhos e nenhum resíduo é enviado.

Observe o perfil dos modos de predição empregados para as seqüências Pedestrian Area, Riverbed, Rush Hour e Sunflower quando o tamanho do quadro é variado de QCIF (176x144 *pixels*) a 1080p (1920x1080 *pixels*), apresentado nas Figuras 3.1 a 3.4.

Aqui cabe uma breve descrição das seqüências de alta resolução empregadas:

- Pedestrian Area (1920x1080 *pixels*, 25 quadros por segundo, progressiva) é uma filmagem de uma calçada. A câmera está parada em uma posição baixa enquanto os pedestres passam.

- Sunflower ( $1920 \times 1080$  pixels, 25 quadros por segundo, progressiva) é uma filmagem bem detalhada. Há uma abelha sobre um girassol e as cores não variam. A câmera está fixa e a cena tem movimento global muito suave.
- Rush-hour ( $1920 \times 1080$  pixels, 25 quadros por segundo, progressiva) é uma filmagem da hora do *rush* em Munique. Vêm-se inúmeros carros movendo-se lentamente. A câmera está fixa.
- Riverbed ( $1920 \times 1080$  pixels, 25 quadros por segundo, progressiva) é uma filmagem de uma espécie de espelho d'água. Sequência de vídeo de compressão bastante difícil.

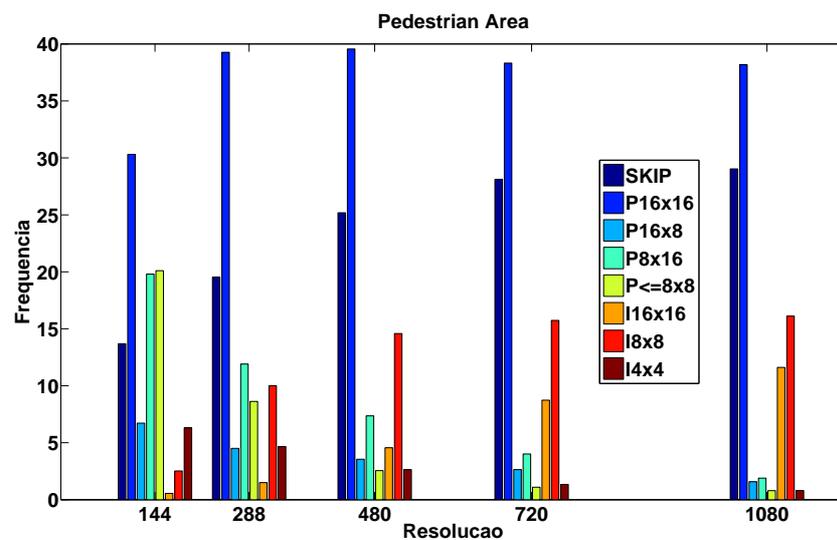


Figura 3.1: Frequência de ocorrência de modos de predição  $\times$  resolução para a seqüência Pedestrian Area.

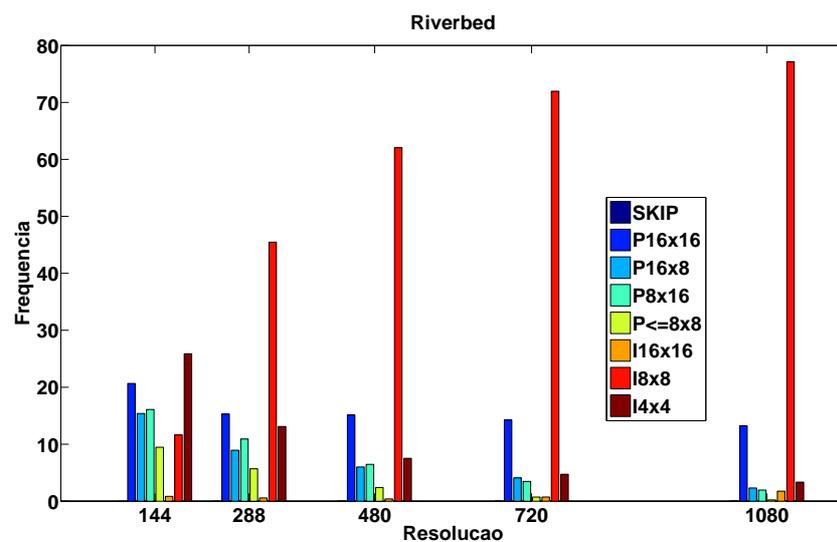


Figura 3.2: Frequência de ocorrência de modos de predição  $\times$  resolução para a seqüência Riverbed.

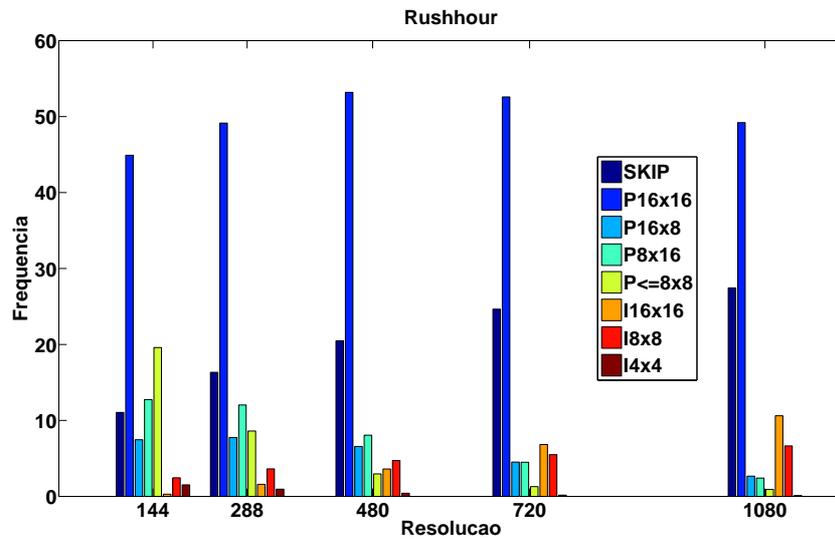


Figura 3.3: Frequência de ocorrência de modos de predição × resolução para a seqüência Rushhour.

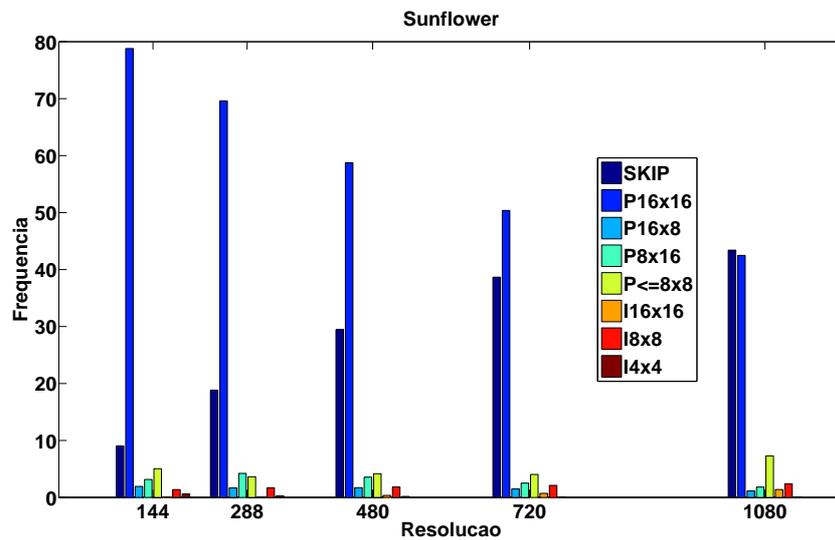


Figura 3.4: Frequência de ocorrência de modos de predição × resolução para a seqüência Sunflower.

Note que os modos tendem a se polarizar em torno de partições de macroblocos maiores à medida que a resolução é aumentada. A única ressalva a ser feita é em relação à seqüência Riverbed, que apresenta um comportamento anômalo por empregar muito o modo de predição Intra-quadro com blocos de  $8 \times 8$  pixels para altas resoluções. Todavia, o comportamento geral sugere que esforço computacional pode ser economizado na codificação de seqüências de vídeo de alta definição evitando-se a predição por compensação de movimentos para macroblocos menores.

Analisar com mais detalhes como são distribuídos os modos de predição para as seqüências HD em diferentes taxas pode nos esclarecer como poderia ser proposta uma metodologia de supressão de modos de predição para a codificação de vídeo de alta resolução. Observe o perfil dos modos de predição empregados para as seqüências HD Pedestrian Area, Riverbed, Rush Hour e Sunflower quando o parâmetro de quantização é variado, apresentado nas Figuras 3.5 a 3.8.

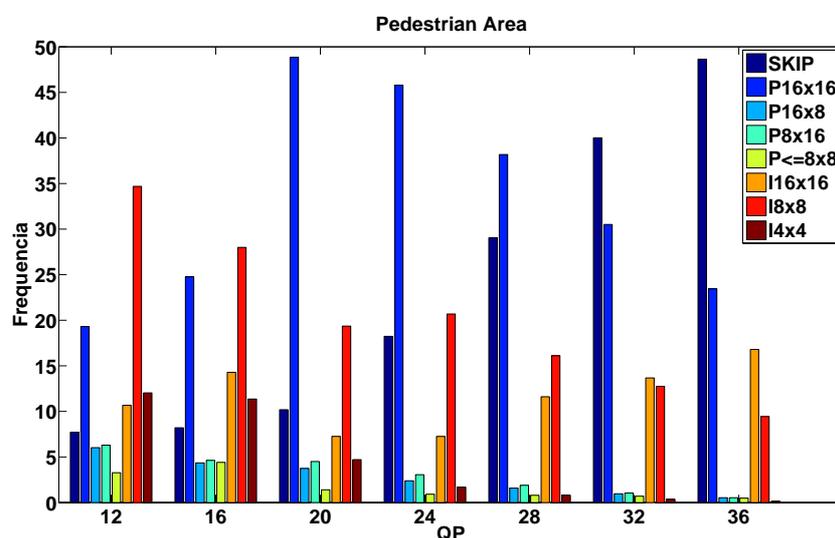


Figura 3.5: Frequência de ocorrência de modos de predição × QP para a seqüência Pedestrian Area.

Como pode ser observado, é comum que poucos modos de predição sejam responsáveis por mais da metade das escolhas na codificação dos macroblocos, mostrando que o módulo de predição do codificador poderia economizar um tempo considerável se soubesse de antemão qual seria o subconjunto de modos **dominantes** para uma dada seqüência de vídeo; além disso, a polarização tende a se acentuar mais na na faixa de valores altos de QP. No Capítulo 5 é apresentado e avaliado um analisador de estatístico que é capaz de determinar os modos mais prováveis e, dessa maneira, reduzir o tempo de codificação de seqüências de vídeo de alta resolução.

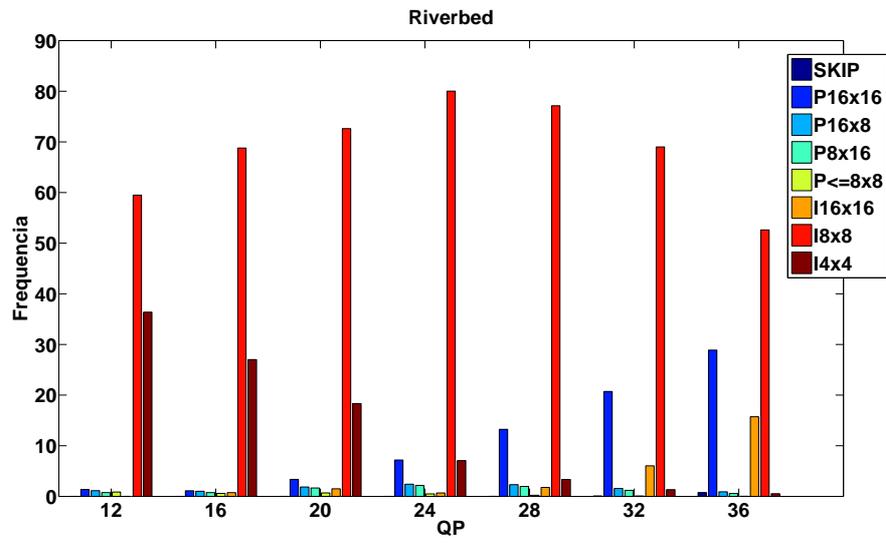


Figura 3.6: Frequência de ocorrência de modos de predição  $\times$  QP para a seqüência Riverbed.

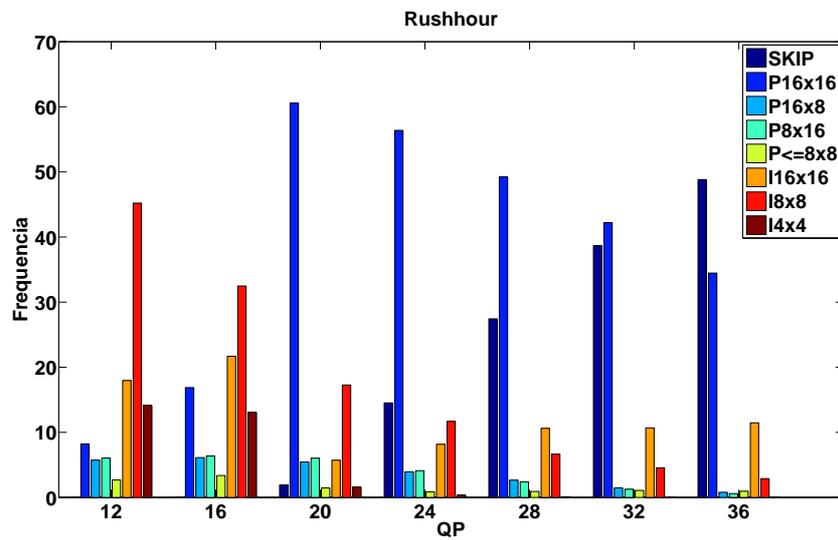


Figura 3.7: Frequência de ocorrência de modos de predição  $\times$  QP para a seqüência Rushhour.

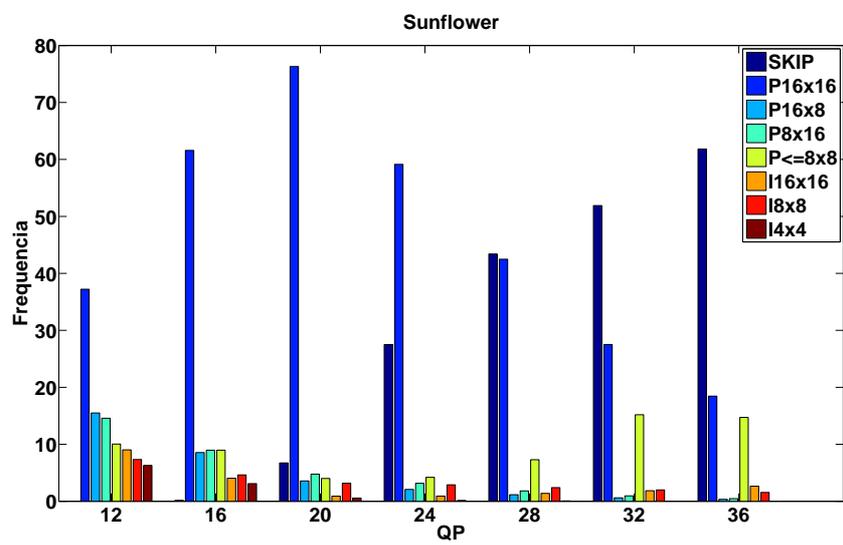


Figura 3.8: Frequência de ocorrência de modos de predição  $\times$  QP para a seqüência Sunflower.



# 4 PREDIÇÃO COM LAÇO ABERTO

## 4.1 DESCRIÇÃO

### 4.1.1 Predição de macroblocos no H.264/AVC

O H.264/AVC é um codificador de vídeo híbrido, isto é, ele é composto por um módulo de transformação e um módulo de predição, interligados por meio de um estágio diferencial e laço de retroalimentação, conforme descrito na Seção 2.3.3. Para realizar predição, um codificador DPCM [4], como qualquer outro sistema de codificação recursivo com perdas, faz o uso de amostras previamente reconstruídas (o codificador deve mimetizar o processo de decodificação) como entrada para o modelo de predição. Dessa maneira, evitam-se incoerências entre os dados disponíveis no codificador e os do decodificador e viabiliza-se reconstrução perfeita. Contudo, o estágio de predição do H.264/AVC é relativamente complexo quando comparado com outros codificadores de vídeo devido a sua gama de modelos de predição que tentam melhor representar o sinal de vídeo.

A Figura 4.1 ilustra um diagrama de blocos simplificado do H.264/AVC com seu estágio de predição destacado. Se levarmos em consideração as técnicas de predição empregadas, é possível dividi-lo em duas partes: a Predição Inter-quadros, em que usam-se modelos temporais e blocos de outros quadros, e a outra chamada Predição Intra-quadro, em que são empregados modelos espaciais e não são feitas referências a blocos de quadros anteriores, somente a blocos do mesmo quadro para predição.

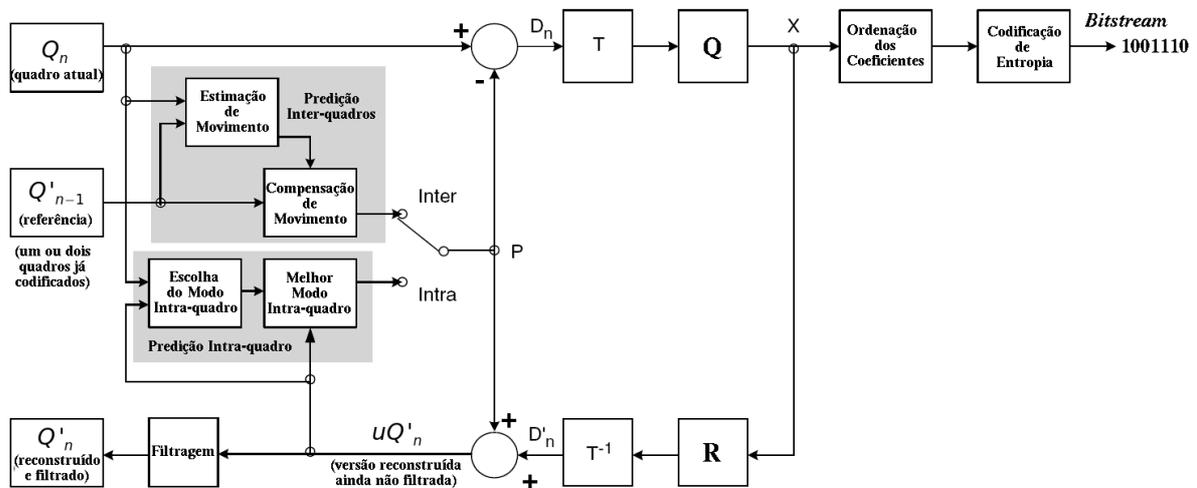


Figura 4.1: Codificador de Vídeo DPCM Híbrido. Adaptado de [2].

Note que, na Figura 4.1, a predição para o macrobloco atual é criada a partir de amostras que já haviam sido codificadas, sejam pertencentes ao mesmo quadro (predição Intra-quadro,  $uQ'_n$ ), sejam de quadros anteriores (predição Inter-quadros,  $Q'_{n-1}$ ). Essa medida evita que ocorra *drifting* no processo de decodificação.

O *drifting* é um problema que ocorre quando o decodificador não consegue reproduzir os mesmos passos usados na codificação devido a descasamento entre os dados reconstruídos ou parâmetros obtidos no decodificador e suas versões aplicadas na codificação. Essa denominação dá-se em função do comportamento tomado pelo decodificador quando enfrenta esse problema: ele parece estar à deriva pois a qualidade do sinal é degradada à medida que o processo de decodificação avança, chegando ao limite em que o sinal decodificado perde sua inteligibilidade. Fontes comuns para esse problema são canal de transmissão com erros ou mesmo codificação/decodificação incorreta do *bitstream*.

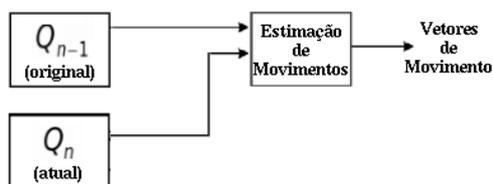
Continuando a explicação da Figura 4.1, no estágio diferencial a predição (Intra-quadro ou Inter-quadros) gera um macrobloco predito que é subtraído do macrobloco atual resultando no resíduo ( $D_n$ ) que é transformado e quantizado ( $X$ ), comprimido e transmitido, juntamente com toda informação necessária (vetores de movimento, tamanho de bloco, direção de extrapolação, etc.) que garanta que o decodificador repita o processo de predição. O decodificador cria uma predição idêntica e adiciona ao resíduo decodificado. Usar sinais previamente codificados e reconstruídos, em vez de sinais originais, como entradas ao processo de predição, garante que as predições calculadas no codificador e no decodificador sejam idênticas.

#### 4.1.2 Predição baseada em dados originais

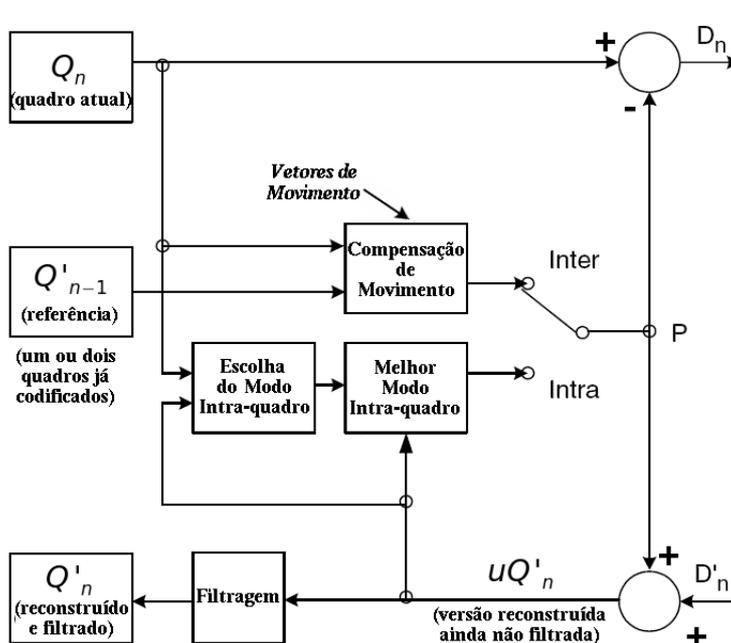
Conforme ilustrado na Figura 4.1, antes de decidir qual é o melhor modo de predição, o codificador realiza uma miríade de testes para escolher entre o melhor modo de predição Inter-quadros ou o melhor modo de predição Intra-quadro. Dada a escolha de um modo de predição, deve-se prover ao decodificador o conjunto de informações que indicam o tamanho do bloco e a direção de extrapolação planar empregada, para o caso Intra, ou o tamanho do bloco e vetores de movimento, para o caso Inter.

Na compressão de vídeo de alta definição, tipicamente pouca distorção é tolerada, o que implica em grande similaridade entre a versão comprimida e a versão original de um vídeo. Portanto, se empregarmos dados originais em vez de reconstruídos nos testes de predição, é provável que o melhor modo de predição escolhido seja o mesmo escolhido usando dados reconstruídos. Por isso, o método proposto, apresentado na Figura 4.2, consiste em usar dados originais para escolher o melhor modo de predição Inter-quadros

mas continua empregando dados localmente decodificados para realizar a compensação de forma a evitar o efeito de deriva (*drifting*) no decodificador. Todos os dados e modos usados para a geração do resíduo estão disponíveis no decodificador. No pior das hipóteses, se o melhor modo encontrado pela aplicação da metodologia proposta for diferente do encontrado pelo “método tradicional”, o único efeito será um consumo de mais bits na codificação do resíduo enquanto a distorção continua ditada pela quantização.



(a)



(b)

Figura 4.2: Processo de predição Inter-quadros (a) paralelizado e nova estrutura do codificador (b) para predição paralelizada.

Conforme indicado na Figura 4.2(a), os testes de predição Inter-quadros dependem tão somente dos dados originais e não mais dos dados reconstruídos conforme mostrado na Figura 4.1. A busca pelo melhor modo de predição Inter-quadros, a etapa com maior complexidade computacional de um codificador H.264/AVC, pode ser destacada do laço de codificação e ser paralelizada dado que não é mais necessário esperar por dados previamente codificados e decodificados, abrindo, de certa maneira, o laço de predição. O

produto do processo da Figura 4.2(a) são os *Vetores de Movimento*, parâmetros necessários para a execução da etapa de *Compensação de Movimentos* da Figura 4.2(b).

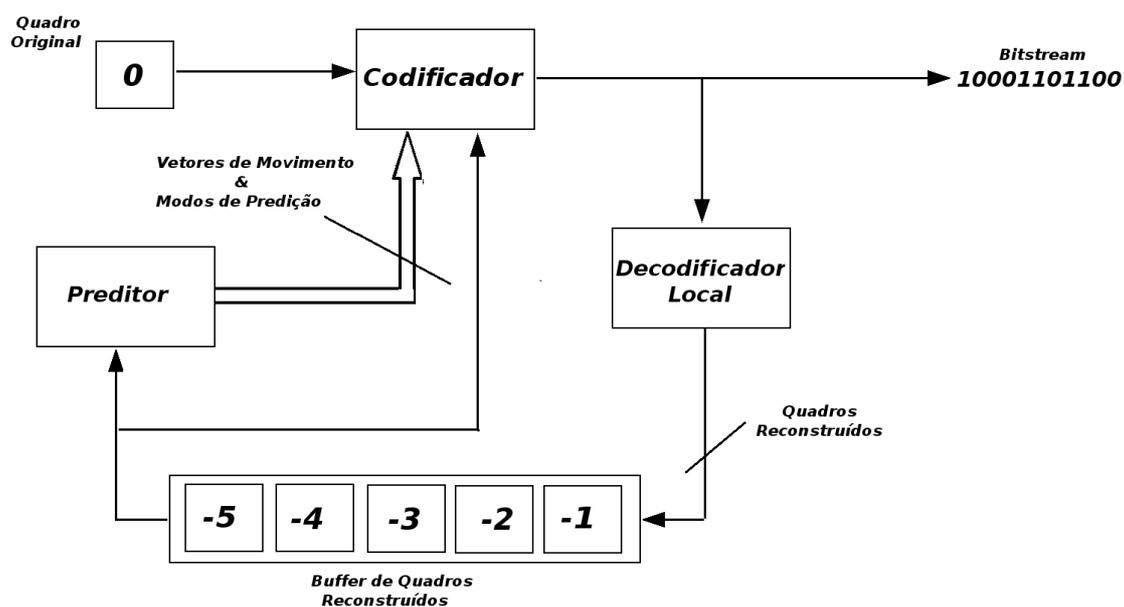
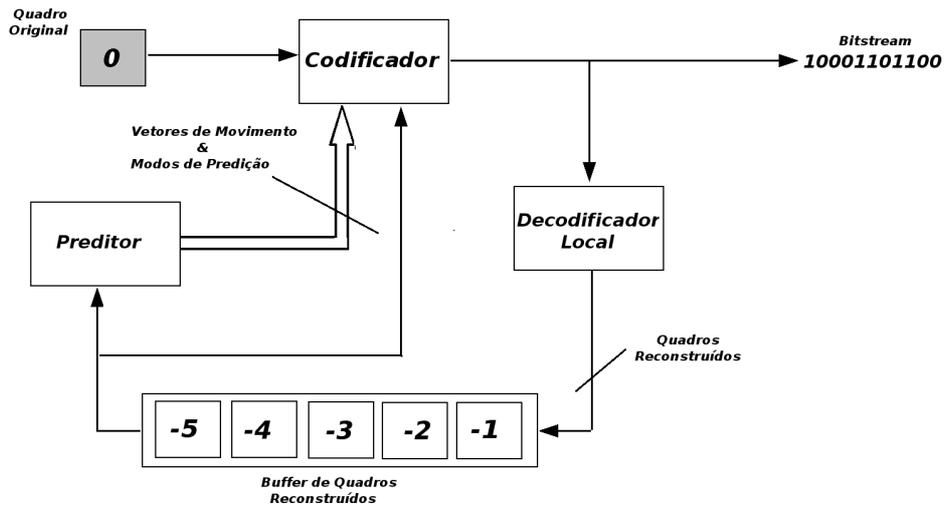


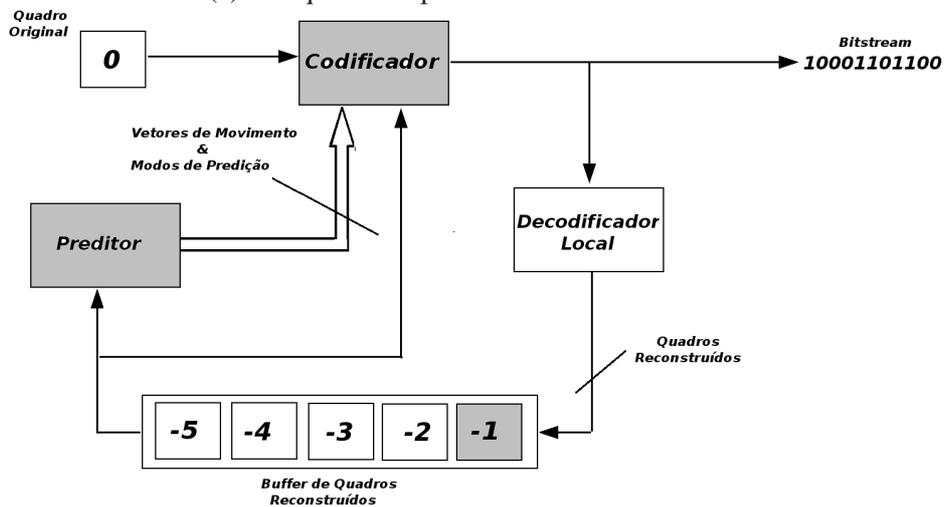
Figura 4.3: Diagrama de previsão original com cinco quadros de referência.

Uma possibilidade a ser explorada por meio dessa técnica consiste na paralelização do processo de previsão Inter-quadros, o que quebra o paradigma de estrutura serializada muito empregada até então nas implementações em *software* de referência de codificadores de vídeo normatizados. Dessa forma, dado um computador pessoal aplicado na realização de testes de previsão, que é capaz de analisar um quadro de referência em  $\Delta t$  segundos, a junção do trabalho de  $n$  desses computadores em paralelo resultará no gasto de  $\Delta t$  segundos para a análise de  $n$  quadros de referência simultaneamente. Isso permitirá a codificação mais rápida de seqüências de vídeo de alta resolução sem a necessidade de empregar *hardware* dedicado.

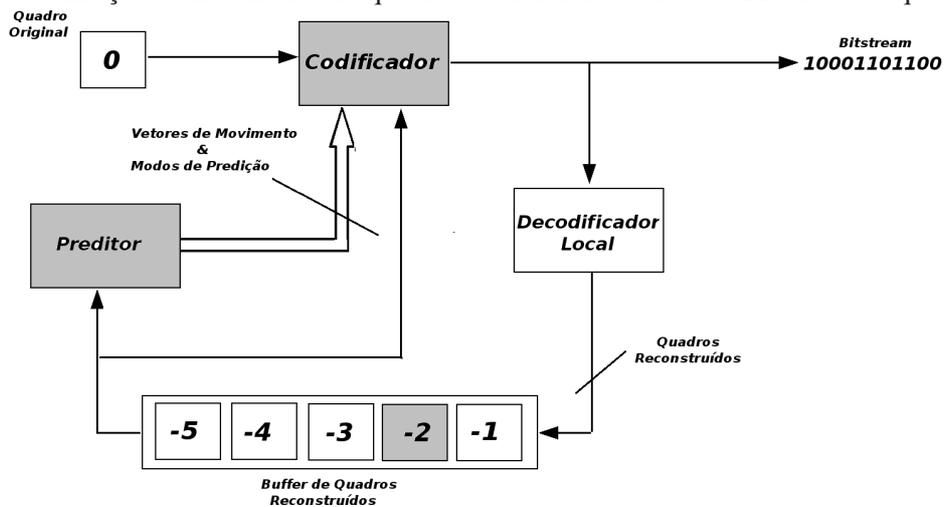
Os benefícios da abertura do laço de codificação podem ser melhor compreendidos por meio da comparação dos procedimentos de codificação original e o processo proposto. A Figura 4.3 apresenta o diagrama do codificador H.264/AVC original somente com seus módulos de interesse representados. As Figuras 4.4(a) a (i) ilustram as etapas de codificação de um quadro a partir de um *buffer* com 5 quadros de referência. Em cada etapa está destacado em cinza qual o módulo “ativo” a cada momento; a atividade de cada etapa enumerada é descrita a seguir:



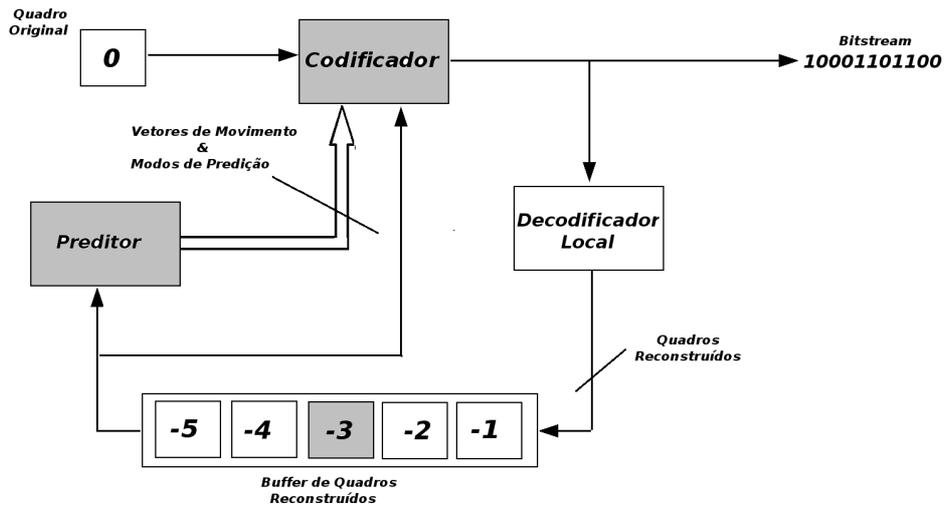
(a) Um quadro é apresentado ao **Codificador**.



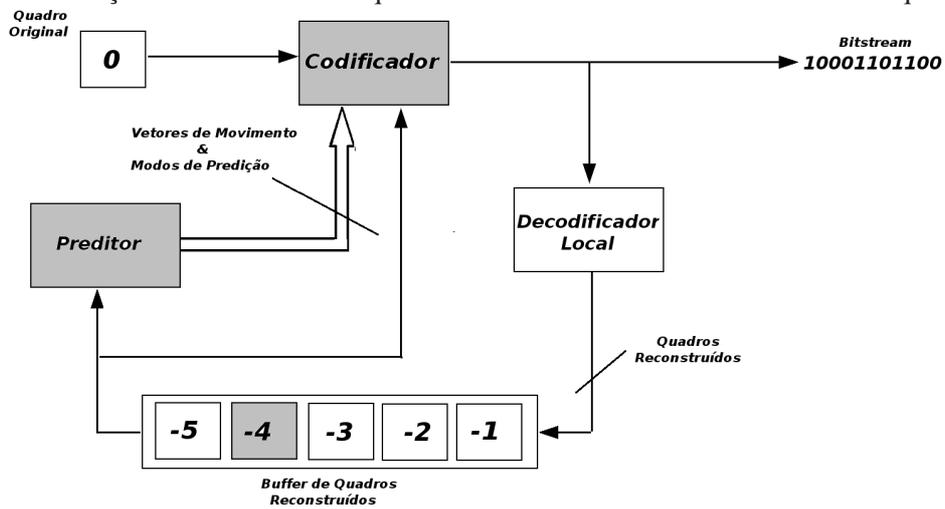
(b) É aplicada estimação de movimento no quadro de referência -1. Note o **Preditor** e o quadro -1 ativos.



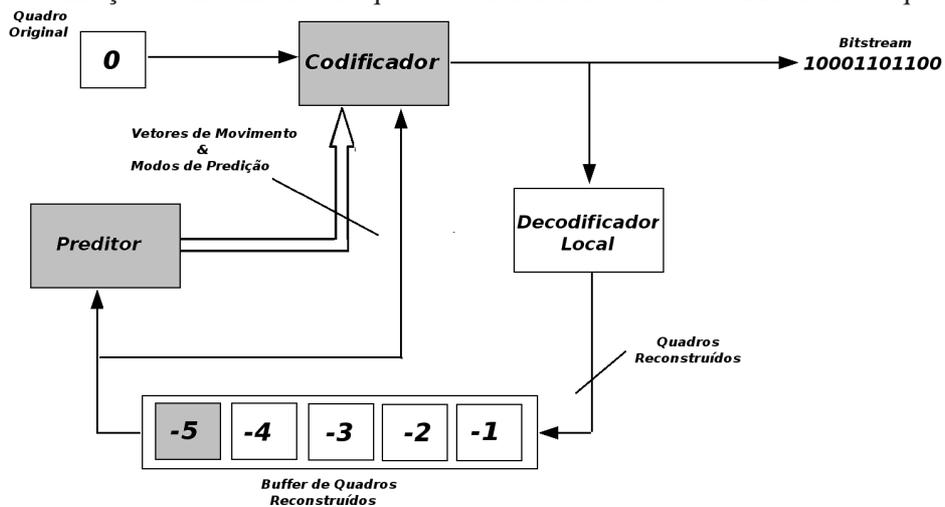
(c) É aplicada estimação de movimento no quadro de referência -2. Note o **Preditor** e o quadro -2 ativos.



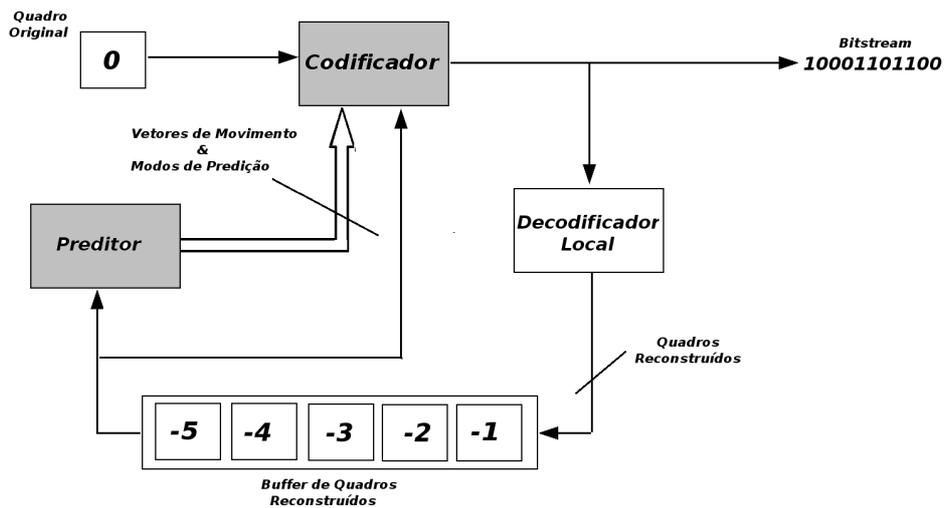
(d) É aplicada estimação de movimento no quadro de referência -3. Note o **Preditor** e o quadro -3 ativos.



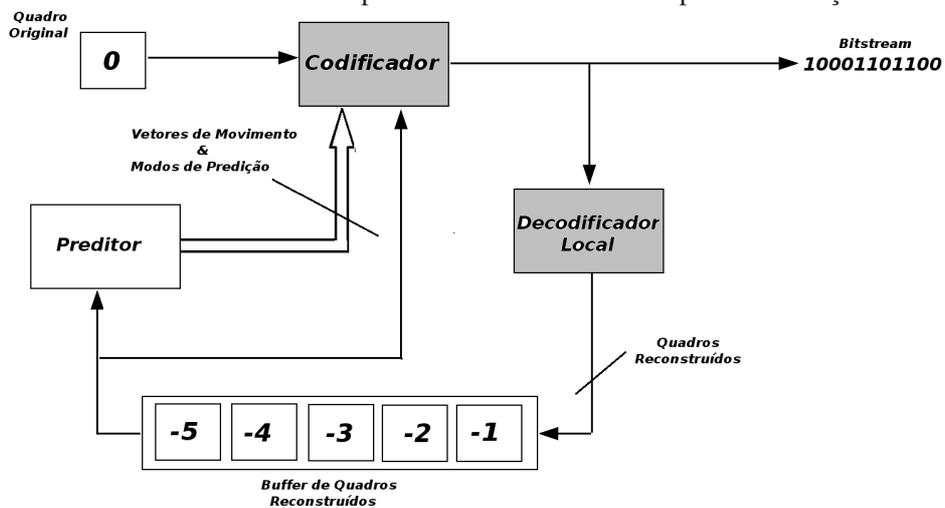
(e) É aplicada estimação de movimento no quadro de referência -4. Note o **Preditor** e o quadro -4 ativos.



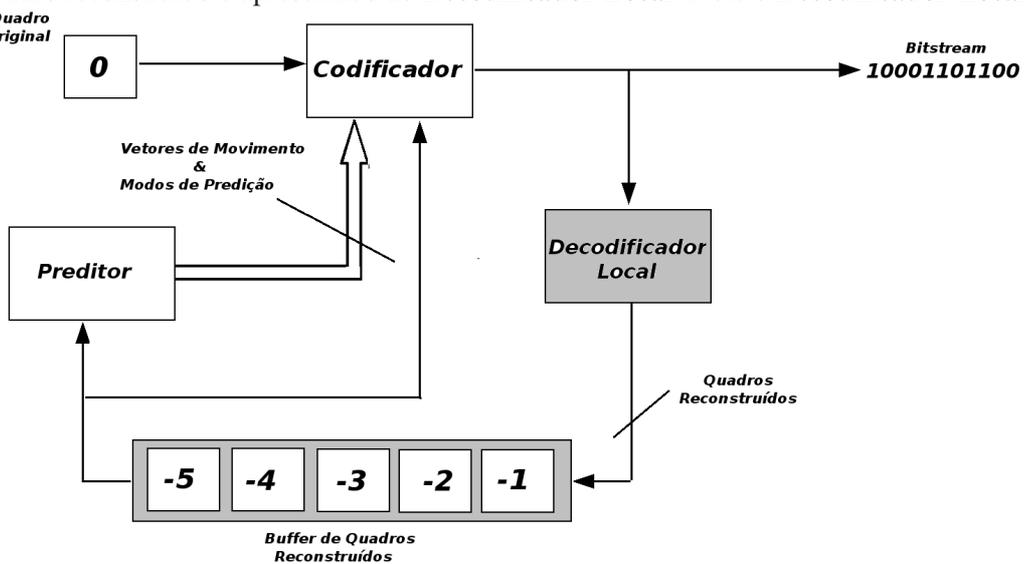
(f) É aplicada estimação de movimento no quadro de referência -5. Note o **Preditor** e o quadro -5 ativos.



(g) As previsões são calculadas no **Preditor** e apresentadas ao codificador para codificação. Note o **Preditor** ativo.



(h) O quadro reconstruído é apresentado ao **Decodificador Local**. Note o **Decodificador Local** ativo.



(i) O **Decodificador Local** atualiza o **Buffer de Referência**. Note o **Buffer de Referência** ativo.

Figura 4.4: Ilustração do processo de previsão original ((a) a (i)) para cinco quadros de referência.

- a. Um quadro é apresentado ao **Codificador**.
- b. É aplicada estimação de movimento no quadro de referência -1 (mais próximo em termos de ordem de codificação).
- c. É aplicada estimação de movimento no quadro de referência -2.
- d. É aplicada estimação de movimento no quadro de referência -3.
- e. É aplicada estimação de movimento no quadro de referência -4.
- f. É aplicada estimação de movimento no quadro de referência -5 (mais distante em termos de ordem de codificação).
- g. As previsões são calculadas no **Preditor** e apresentadas ao codificador que decide o melhor modo e realiza a codificação.
- h. O quadro reconstruído é apresentado no **Decodificador Local** que
- i. atualiza o **Buffer de Quadros Reconstruídos**, também denominado **Buffer de Referência**.

Note que entre os itens (c) e (g) cada um dos quadros de referência é analisado sequencialmente para gerar o *bitstream* do quadro de entrada.

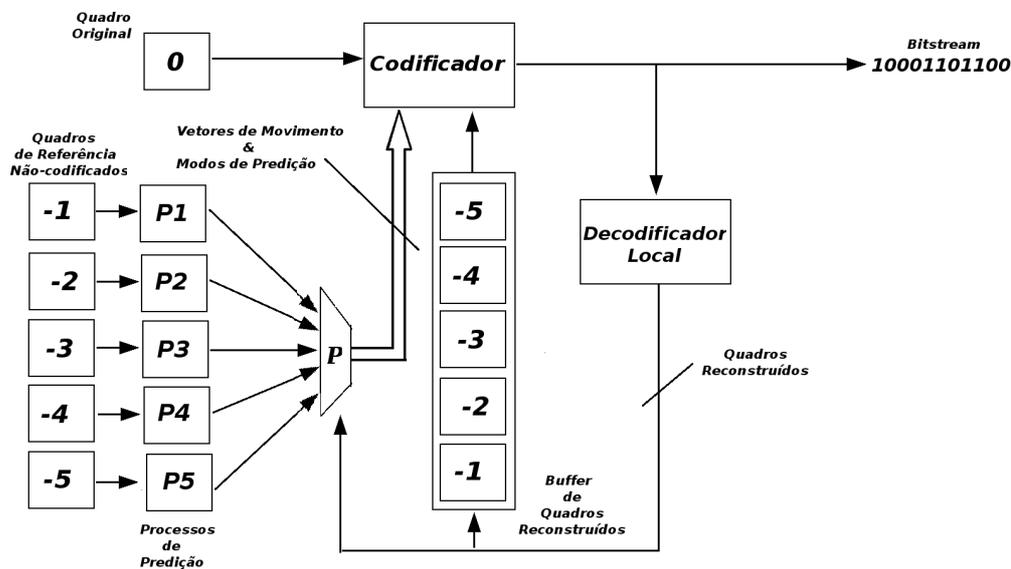
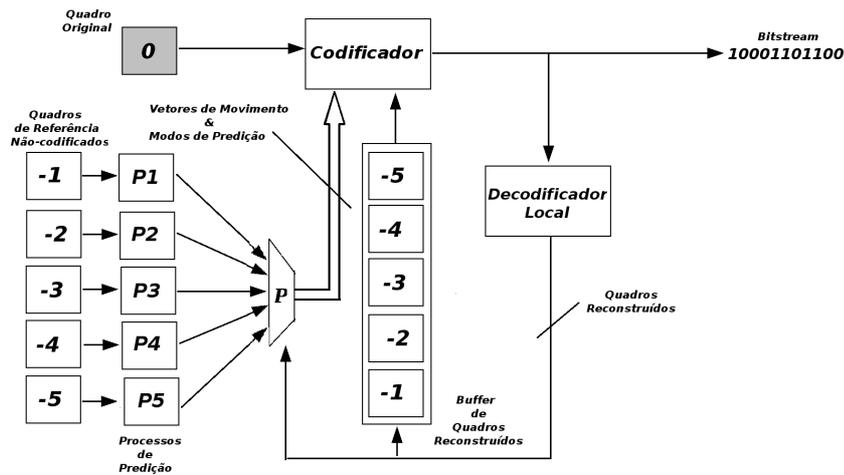
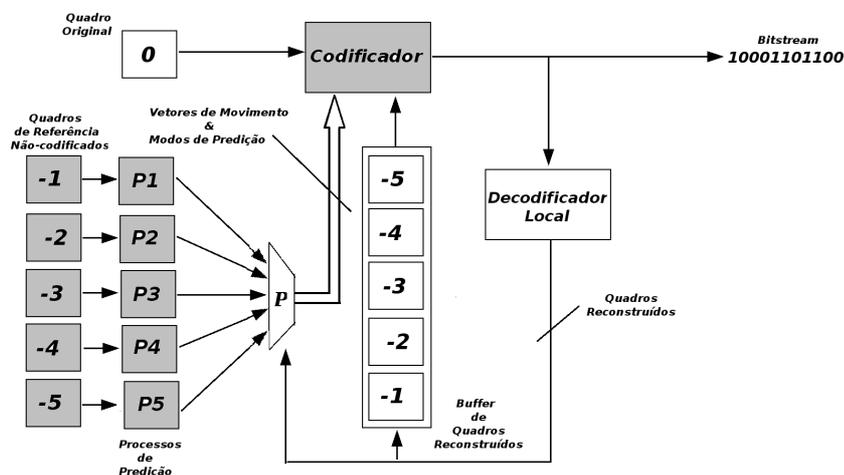


Figura 4.5: Diagrama de predição proposto com cinco quadros de referência.

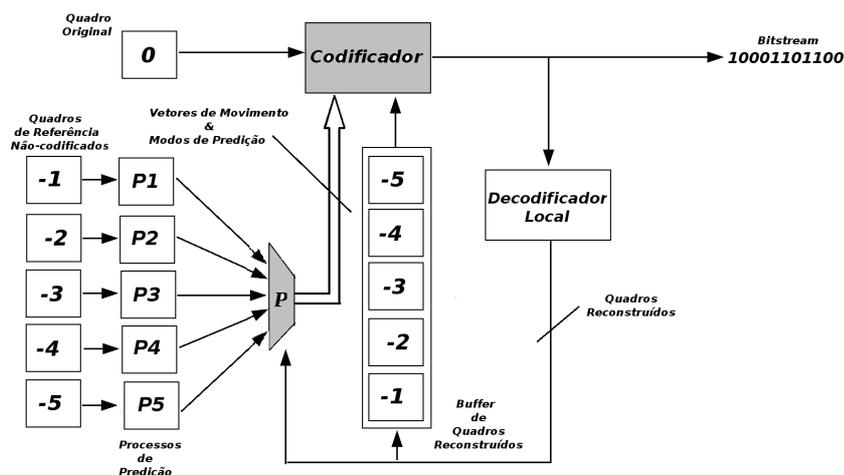
O processo proposto, Figura 4.5, realiza a codificação de forma diferente como pode ser acompanhado pelas Figuras 4.6(a) a (e) e possibilita uma redução significativa do tempo empregado na etapa de



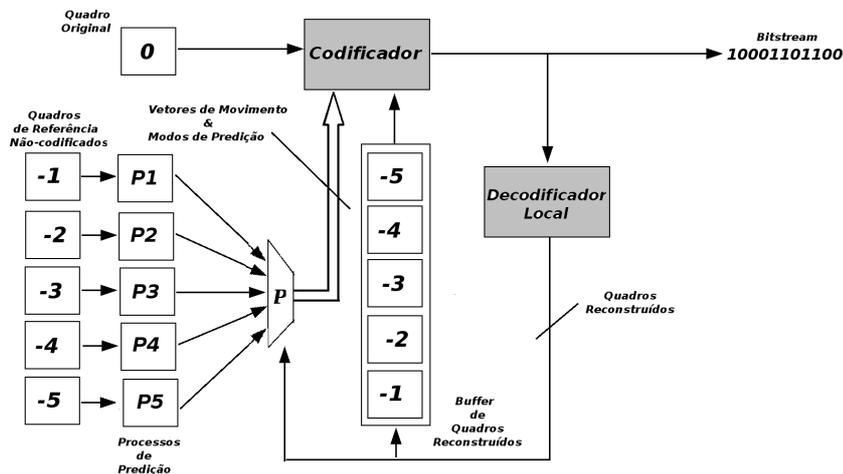
(a) Um quadro é apresentado ao **Codificador**.



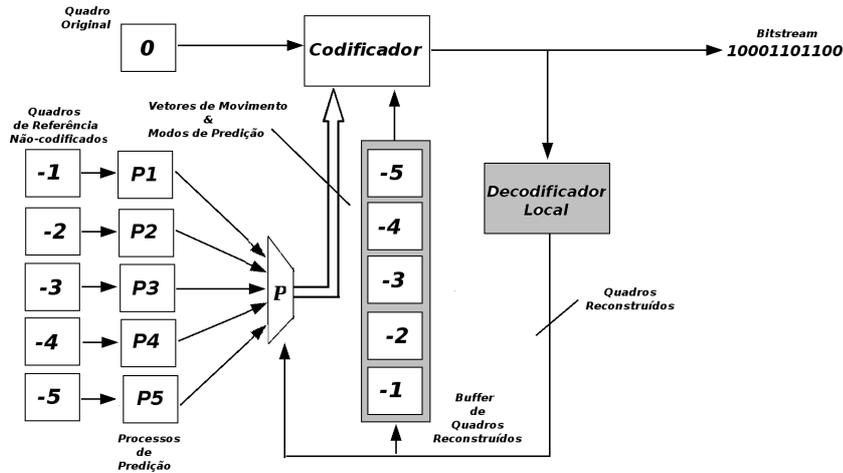
(b) É aplicada estimação de movimento aos cinco quadros de referência simultaneamente. Note **P1** a **P5** ativos.



(c) As previsões são calculadas no preditor **P** e apresentadas ao codificador. Note **P** ativo.



(d) O quadro é apresentado ao **Decodificador Local**. Note o **Decodificador Local** ativo.



(e) O **Decodificador Local** reconstrói o quadro e atualiza o **Buffer de Referência**. Note o **Buffer de Referência** ativo.

Figura 4.6: Processo de predição proposto ( ( a ) a ( e ) ) para cinco quadros de referência.

predição Inter-quadros por meio de paralelização. Seguem abaixo a seqüência de atividades realizadas pelo codificador modificado:

- a. Um quadro é apresentado ao **Codificador**.
- b. É aplicada estimação de movimento aos cinco quadros de referência simultaneamente. **P1 a P5** são processos de predição por estimação de movimentos que são executados em cinco máquinas de processamento distintas.
- c. As predições são calculadas no preditor **P** e apresentadas ao codificador.
- d. O quadro é apresentado ao **Decodificador Local**.
- e. O **Decodificador Local** reconstrói o quadro e atualiza o *buffer* de referência.

## 4.2 RESULTADOS EXPERIMENTAIS

Partindo do *software* de referência do padrão, o JM-10.2 [43], aplicou-se a metodologia de predição sobre dados originais ao H.264; cabe ressaltar que neste trabalho não foi implementado o estágio de predição paralelizado. Foram usadas as seqüências de vídeo HD Pedestrian Area, Riverbed, Rush-hour e Sunflower.

De cada seqüência de vídeo, foram usados os primeiros 20 quadros e curvas de desempenho taxa  $\times$  distorção foram traçadas por meio da variação do parâmetro de quantização (QP) no intervalo de 8 a 36. Nesse intervalo de QP são atingidos critérios de qualidade e de taxa de transmissão para vídeo de alta definição. Nas Figuras 4.7 a 4.10 são apresentados gráficos compostos por duas curvas: uma curva de desempenho para o codificador original (JM10.2) e outra curva para o codificador modificado (JM10.2-OL). A diferença média de PSNR entre as duas curvas de taxa  $\times$  distorção foi calculada de acordo com a metodologia proposta em [44]. Seu autor sugere o levantamento de curvas taxa  $\times$  distorção com quatro pontos, e, para cada uma delas, deverá ser calculada a integral; a diferença média de desempenho entre as curvas é tomada como a diferença das integrais dividida pelo intervalo de integração.

A técnica proposta foi testada em outras condições para a verificação da possibilidade de generalização para resoluções menores. As Figuras 4.11 a 4.13 apresentam a diferença de desempenho do codificador proposto quando o cenário é a compressão de seqüências de resolução CIF (352 $\times$ 288 *pixels*).

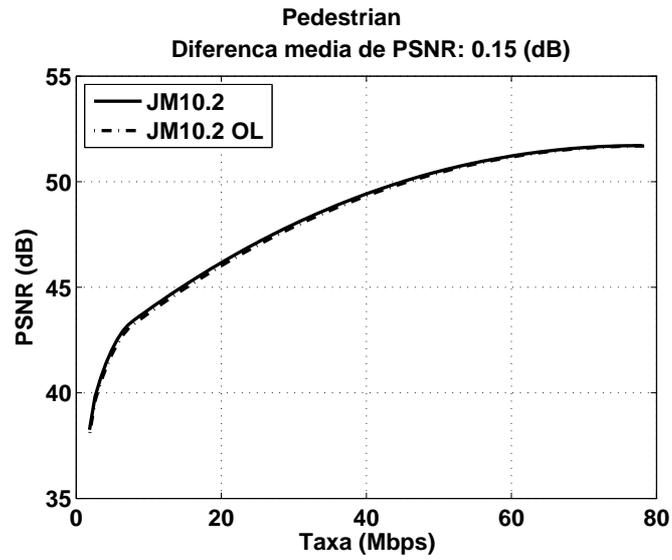


Figura 4.7: Curva de taxa  $\times$  distorção para a seqüência Pedestrian Area: JM10.2  $\times$  JM10.2-OL.

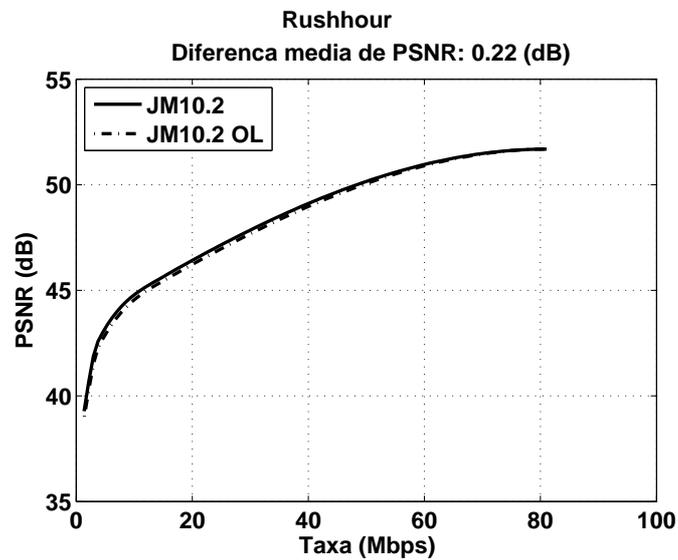


Figura 4.8: Curva de taxa  $\times$  distorção para a seqüência Rushhour: JM10.2  $\times$  JM10.2-OL.

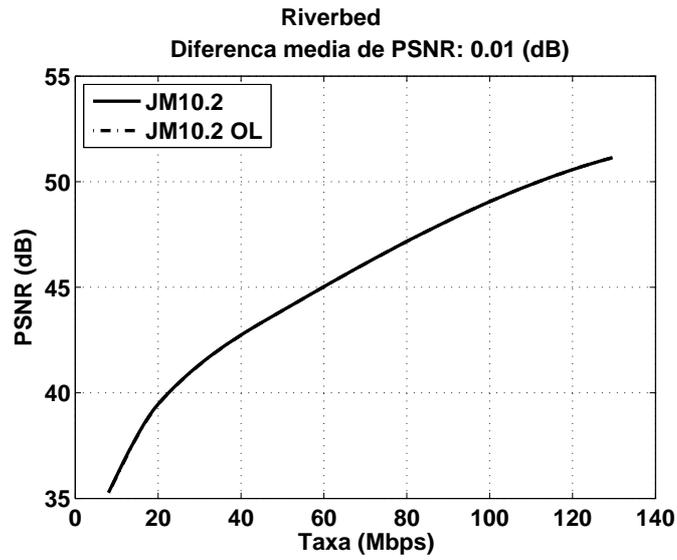


Figura 4.9: Curva de taxa  $\times$  distorção para a seqüência Riverbed: JM10.2  $\times$  JM10.2-OL.

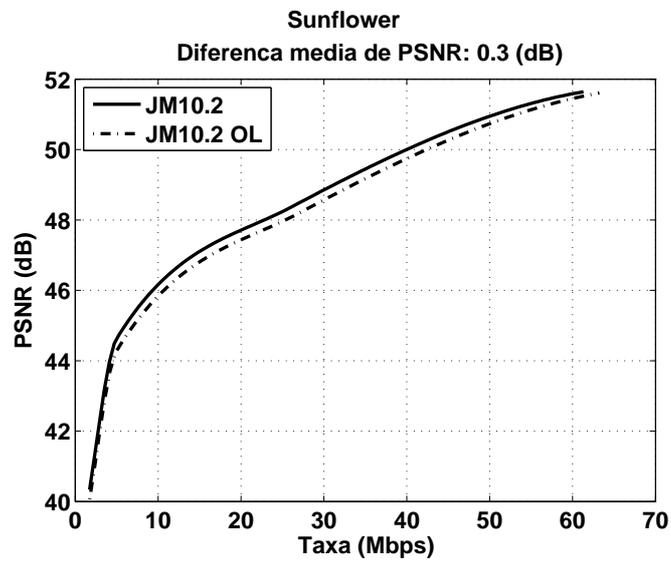


Figura 4.10: Curva de taxa  $\times$  distorção para a seqüência Sunflower: JM10.2  $\times$  JM10.2-OL.

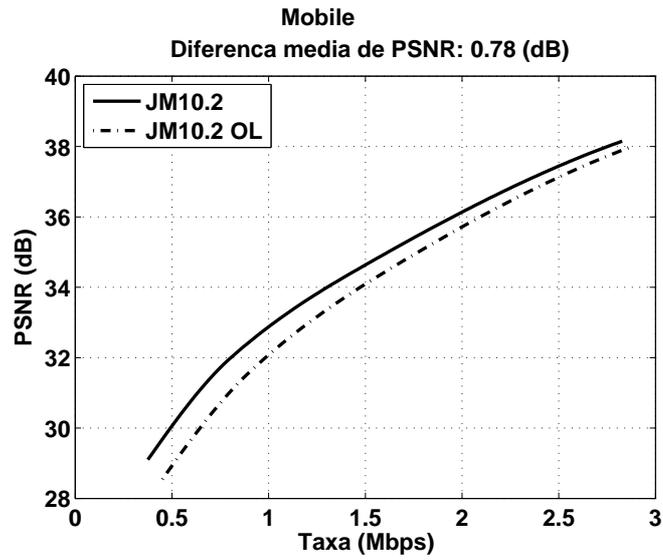


Figura 4.11: Curva de taxa  $\times$  distorção para a seqüência CIF Mobile: JM10.2  $\times$  JM10.2-OL.

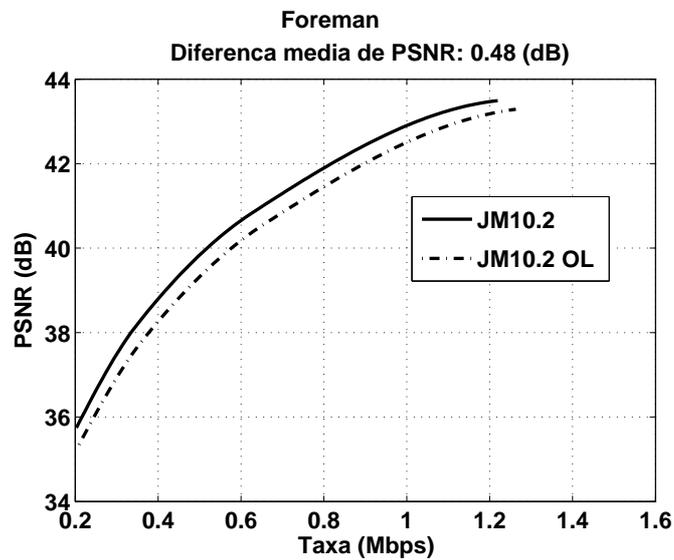


Figura 4.12: Curva de taxa  $\times$  distorção para a seqüência CIF Foreman: JM10.2  $\times$  JM10.2-OL.

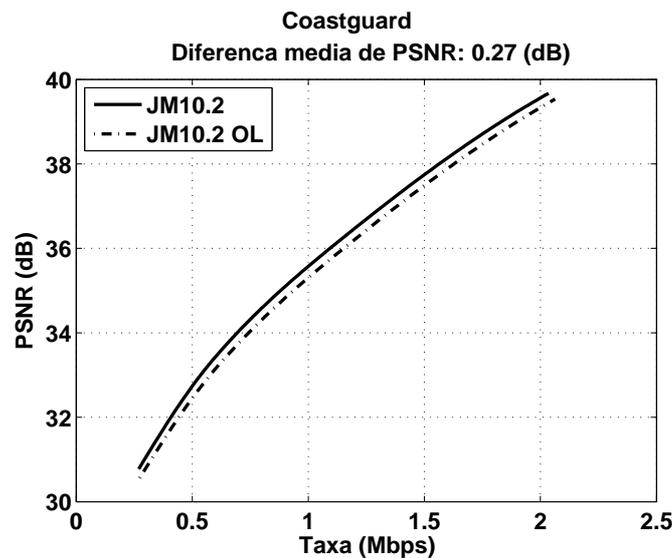


Figura 4.13: Curva de taxa × distorção para a sequência CIF Coastguard: JM10.2 × JM10.2-OL.

### 4.3 DISCUSSÃO

As curvas apresentam uma pequena perda de desempenho quando são aplicados dados originais em vez de dados reconstruídos no processo de predição do H.264/AVC. A diferença é justificada pelos eventuais descasamentos entre o melhor modo de predição encontrado por cada uma das metodologias.

Em altas taxas, região em que o parâmetro de quantização (QP) apresenta valores pequenos, as curvas se encontram o que confirma o fato de que, quanto menor a distorção contida na versão codificada, mais próximas serão as predições sobre dados originais e dados reconstruídos.

As perdas de desempenho observadas são justificadas devido aos diferentes vetores de movimento encontrados pela metodologia, o que garante compressão do sinal com ganhos de codificação ligeiramente menores pois é necessário o gasto de mais taxa para o envio da sequência de vídeo codificada. Manter o laço de predição possibilita explorar melhor os contextos para a predição, implica em melhor compressão em termos taxa × distorção e garante compressão do sinal sem *drifting*; contudo apresenta o inconveniente da serialização do processo preditivo.

A técnica aqui proposta pôde ser avaliada em outras condições (Figuras 4.11 a 4.13) para verificação da possibilidade de generalização da técnica. Observou-se perdas de desempenho significativas na aplicação da técnica para a codificação de sequências no formato CIF. Isso pode ser justificado devido aos descasamentos entre os modos de predição e vetores de movimento encontrados pela metodologia proposta quando comparados ao proposto na norma do padrão; esses descasamentos resultaram em codificação

menos eficiente em termos de taxa  $\times$  distorção, pois a codificação pela metodologia proposta consumiu mais bits na representação dos sinais de vídeo.

Os resultados das Figuras 4.7 a 4.10 mostram que paralelizar o processo de predição apresenta-se como uma alternativa para a viabilização da compressão de vídeo de alta resolução em tempo real, uma tarefa que costuma ser demorada devido ao elevado volume de dados a ser analisado. A paralelização surge, portanto, como uma quebra de paradigma de codificação em *softwares* de referência no momento em que se abre o laço de predição para o codificador DPCM e se garante codificação sem *drifting*. Cabe ressaltar que a predição com laço aberto não procura reduzir a complexidade de codificação da seqüência de vídeo, mas sim usar vários sistemas em paralelo para realizar a tarefa de codificação de uma maneira mais rápida; além disso, o intuito deste estudo foi verificar a viabilidade do emprego de predição com dados originais, metodologia que abre oportunidade para paralelização do processo de codificação, não implementada neste trabalho.

# 5 MÓDULO ANALISADOR ESTATÍSTICO PARA REDUÇÃO DE COMPLEXIDADE DE PREDIÇÃO

## 5.1 DESCRIÇÃO

Uma forma de tornar o codificador H.264/AVC menos complexo computacionalmente seria evitar testes de predição desnecessários, descartando testar modos de codificação com baixa probabilidade de ocorrência em uma seqüência de vídeo. A questão a ser levantada é: como saber previamente quais os modos a serem testados?

Empregar técnicas de amostragem estatística é uma alternativa. Por exemplo, poderíamos codificar um quadro P numa seqüência de vídeo e supor que os modos dominantes encontrados sejam representativos para toda a seqüência. Porém o sinal de vídeo não tem esta característica de estacionariedade.

Ao tomar as estatísticas do quadro anterior no levantamento do subconjunto de modos de predição do quadro atual teremos uma boa idéia de quais serão os modos dominantes sem precisar codificar completamente o quadro atual. Recalcular essas estatísticas a cada quadro parece uma abordagem melhor, mesmo assim o resultado ainda não é exato.

A metodologia proposta para determinação dos modos de predição ditos **dominantes** é ilustrada na Figura 5.1. Inicia-se o procedimento por meio da seleção de um subconjunto de macroblocos, cujo tamanho é discutido na Seção 5.1.1. Para compor o subconjunto, optou-se por amostrar macroblocos em posições aleatórias e submetê-los a processo normal de predição na codificação. Terminada a codificação do subconjunto amostral, calcula-se a freqüência de ocorrência de cada um dos modos de predição listados na Seção 3.3, à exceção do modo SKIP, e ordenam-se decrescentemente os modos de predição. A última etapa é a determinação dos participantes do subconjunto dos modos dominantes, cuja composição começa com o modo mais frequente e continua com a adição outros modos até que a freqüência acumulada dos eleitos seja amplamente majoritária, por exemplo, que supere 80% dos casos.

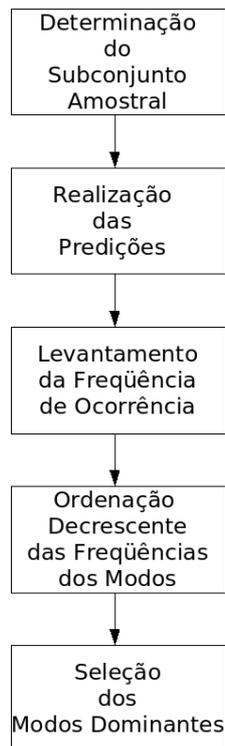


Figura 5.1: Fluxograma da metodologia de determinação dos modos de predição dominantes.

### 5.1.1 Avaliação do tamanho do subconjunto de macroblocos para determinação dos modos dominantes de um quadro

A determinação do tamanho do subconjunto de macroblocos usados para encontrar os modos **dominantes** para uma dada seqüência deve obedecer a critérios estatísticos de forma a empregar a quantidade correta de indivíduos (aqui macroblocos) necessários para análise de forma a se obter parâmetros.

Geralmente, pesquisas são realizadas através de estudo dos elementos que compõem uma amostra extraída da população que se pretende analisar. É compreensível que o estudo de todos os elementos da população possibilita conhecimento preciso das variáveis que estão sendo pesquisadas; todavia nem sempre é possível obter as informações de todos os elementos da população. Limitações de tempo, custo e as vantagens do uso das técnicas estatísticas de inferências justifica o uso de planos amostrais. [45]

A teoria de amostragem estatística disponibiliza ferramentas para se determinar o tamanho conveniente de uma amostra de  $x$  indivíduos para estimar uma certa característica de uma população de  $N$  indivíduos. Neste trabalho, estamos interessados na proporção  $p$  de elementos que possuem certo atributo, especificamente a proporção  $p$  de indivíduos da população que pertencem ao subconjunto dos modos ditos **dominantes**. Usualmente se parte de uma estimativa  $\hat{p}$  para o valor da proporção  $p$  que se quer avaliar.

Em modelos de estimação de características de populações, é necessário apontar qual será o erro amostral  $d$  e o nível de confiança para a metodologia. O erro amostral  $d$  é o máximo desvio tolerado entre a estimativa  $\hat{p}$  e o verdadeiro valor da proporção  $p$ , cujo valor não é conhecido. Ele define uma faixa de valores que englobará o real valor da estatística  $p$ . A abordagem mais simples supõe que a proporção  $p$  a ser estimada assume uma distribuição normal [46] e exige o estabelecimento de um nível de confiança que será uma medida do quão confiável será a estimativa obtida. Especificados esses parâmetros, a teoria de estatística amostral disponibiliza o modelo [45] da Equação 5.1 para o dimensionamento do tamanho da amostra  $x$ :

$$x = \frac{NZ^2\hat{p}(1 - \hat{p})}{(N - 1)d^2 + Z^2\hat{p}(1 - \hat{p})} \quad (5.1)$$

onde  $Z$  é o valor da abscissa da curva normal padrão para o nível de confiança estabelecido. Por exemplo, para o nível de confiança de 99%,  $Z = 2,57$ .

Um exemplo permitirá esclarecer o procedimento de determinação do tamanho do conjunto amostral. Suponha que a variável escolhida num estudo seja a proporção de eleitores favoráveis ao candidato  $X$  e que o investigador tenha elementos para suspeitar que essa percentagem seja de 30%. Admita a população finita de  $N = 20000$  eleitores e que se deseje um nível de confiança de 99%, isto é, presume-se que 99% das vezes a estimativa  $\hat{p}$  encontra-se dentro do intervalo cujo tamanho é definido pelo valor do erro amostral  $d$ , tomado como 2%. Assim,

$$\begin{aligned} Z &= 2,57, \\ \hat{p} &= 0,30, \\ N &= 20000, \\ d &= 0,02, \\ x &= \frac{20000 \times 2,57^2 \times 0,30(1 - 0,30)}{(20000 - 1) \times 0,02^2 + 2,57^2 \times 0,30(1 - 0,30)} = 2956. \end{aligned}$$

Ou seja, 2956 eleitores serão necessários para que a pesquisa atenda aos critérios estabelecidos.

O analisador estatístico proposto neste trabalho lançará mão de um processo de amostragem durante a etapa de *Determinação do Subconjunto Amostral* do fluxograma da Figura 5.1, o que nos faz aplicar o modelo da Equação 5.1 para balizar a determinação da quantidade  $x$  de macroblocos mais adequada para a análise estatística a ser realizada. Como o foco é a codificação de seqüências HD 1080p (1920×1080 *pixels*), que são compostas por 8160 macroblocos, o tamanho da população é  $N = 8160$ . Estamos interessados na proporção  $p$  de elementos que possuem certo atributo, especificamente a proporção  $p$  de

indivíduos da população que pertencem ao subconjunto dos modos ditos **dominantes**. Partimos da hipótese que esse subconjunto responda por 80% dos indivíduos, macroblocos de um quadro; portanto  $\hat{p} = 0,8$ . Usando um desvio amostral  $d$  de 2% e um nível de confiança de 99%, o modelo da Equação 5.1 indica que serão necessárias  $x = 1996$  amostras para compor o subconjunto amostral.

Como se espera que a frequência relativa acumulada dos modos **dominantes** supere 80% dos casos, na contagem acumulativa esse valor pode superar os 80% esperados; faz-se necessário avaliar se uma amostra de  $x = 1996$  macroblocos ainda é conveniente para esses casos.

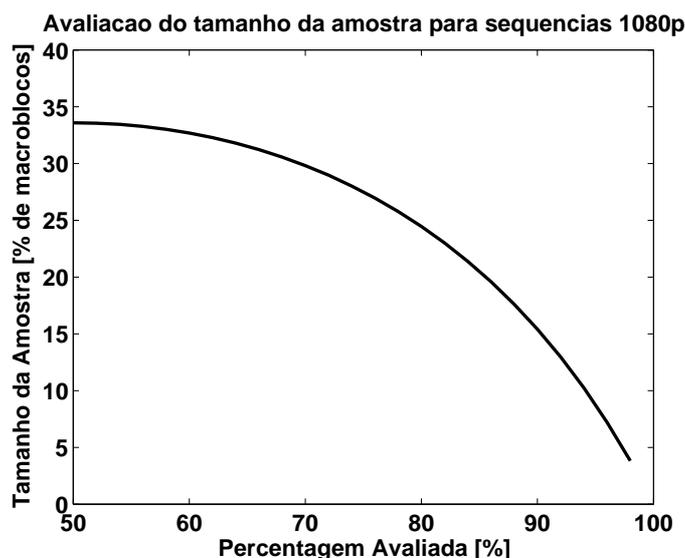


Figura 5.2: Variação do tamanho da amostra *versus* proporção da população a ser verificada.

Um gráfico que apresente o tamanho da amostra *versus* a proporção da população a ser verificada é útil para esse fim. Como pode ser verificado da Figura 5.2, a curva do tamanho da amostra *versus* proporção da população é decrescente, confirmando que o valor encontrado de 1996 (que corresponde a 24,46% dos macroblocos de um quadro 1080p) ficará superdimensionado caso a proporção dos modos dominantes englobe mais que 80% dos macroblocos do quadro.

### 5.1.2 Avaliação do erro médio na predição dos modos dominantes do quadro atual a partir do subconjunto amostral.

Os modos dominantes de cada quadro foram contrastados com os modos do quadro precedente para avaliar a metodologia. Para evitar esforço computacional elevado, usamos como estratégia tomar um subconjunto dos macroblocos de cada quadro. A fim de determinar qual o tamanho adequado de um subconjunto, fizemos testes em que avaliaram-se os acertos na predição dos modos dominantes para seis

tamanhos diferentes do subconjunto amostral:

- para 10% dos macroblocos,
- para 15% dos macroblocos,
- para 20% dos macroblocos,
- para 25% dos macroblocos (quantidade ideal),
- para 30% dos macroblocos e
- para 35% dos macroblocos;

e levantamos a taxa de erro médio (modos errados por quadro) na determinação dos modos dominantes para cada uma das quatro seqüências de vídeo em alta definição empregadas nesse trabalho. Os resultados obtidos são apresentados nas Figuras 5.3 a 5.6. De cada seqüência optou-se por codificar 50 quadros com o QP sendo variado entre 12 a 36.

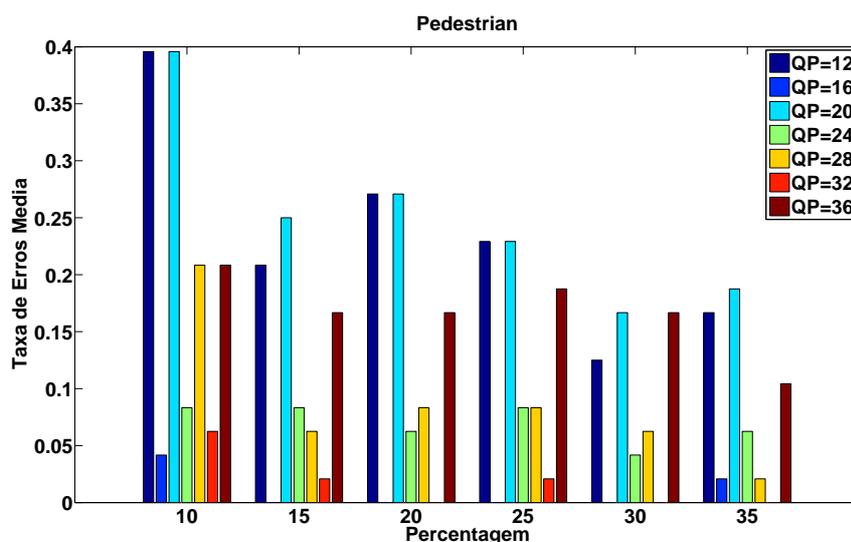


Figura 5.3: Taxa de erro médio na predição dos modos de predição dominantes do quadro atual  $\times$  porcentagem de macroblocos onde são avaliados todos os modos para a seqüência Pedestrian Area.

A observação das Figuras 5.3 a 5.6 nos indica que a taxa de erros na predição dos modos dominantes tende a diminuir com o aumento do tamanho do subconjunto de macroblocos empregado na predição e assume valores menores quando o QP é grande, contudo não é possível afirmar que ela tende a diminuir com o aumento do QP. Estes resultados mostram que aumentar o subconjunto amostral além de 25% não impacta muito no desempenho da amostragem. Fato interessante a se notar é que seqüências em que a

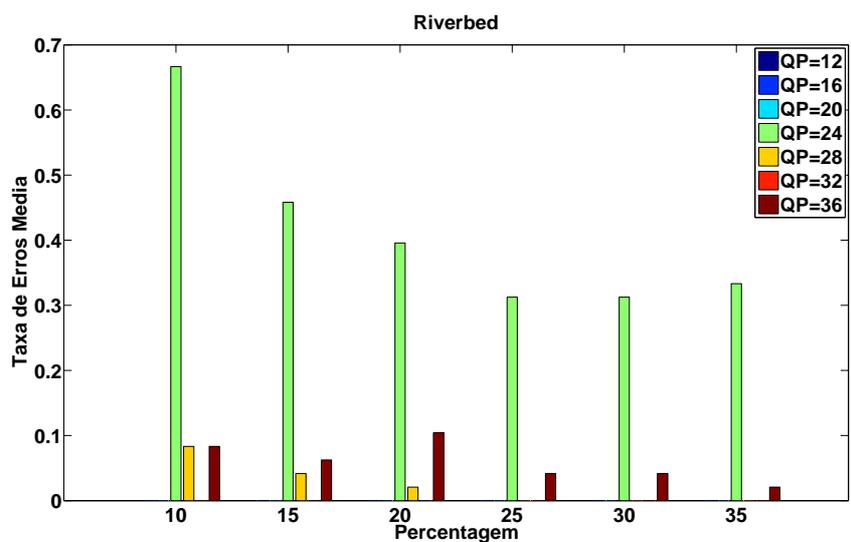


Figura 5.4: Taxa de erro médio na predição dos modos de predição dominantes do quadro atual  $\times$  porcentagem de macroblocos onde são avaliados todos os modos para a seqüência Riverbed.

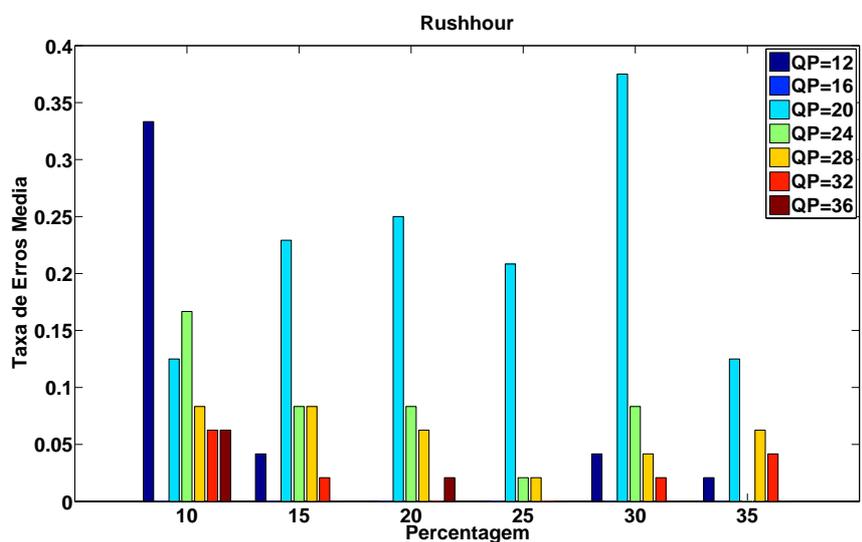


Figura 5.5: Taxa de erro médio na predição dos modos de predição dominantes do quadro atual  $\times$  porcentagem de macroblocos onde são avaliados todos os modos para a seqüência Rushhour.

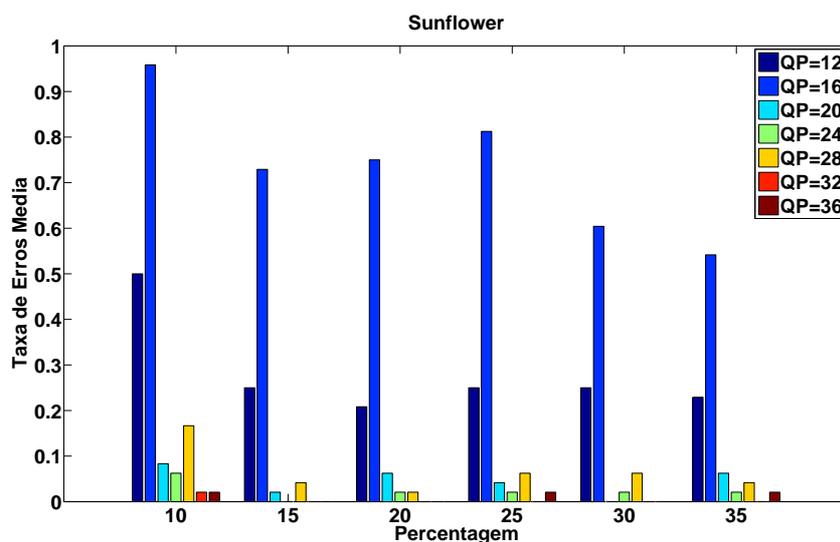


Figura 5.6: Taxa de erro médio na predição dos modos de predição dominantes do quadro atual  $\times$  porcentagem de macroblocos onde são avaliados todos os modos para a seqüência Sunflower.

estimação de movimentos não funciona bem em termos de taxa  $\times$  distorção, como é o caso de Riverbed, têm seus modos de predição bastante polarizados em um pequeno subconjunto e apresenta baixa taxa de erro no levantamento da estatística.

### 5.1.3 Avaliação do erro médio na predição dos modos dominantes do próximo quadro a partir do subconjunto amostral.

Uma vez observado o comportamento da predição dos modos dominantes do quadro atual a partir de um subconjunto amostral, avaliaremos agora como se comporta o erro na predição dos modos dominantes do próximo quadro a partir de diferentes tamanhos do subconjunto amostral do quadro anterior.

Empregou-se os seis tamanhos de subconjunto amostral da Seção 5.1.2 e os resultados obtidos são apresentados nas Figuras 5.7 a 5.10. De cada seqüência optou-se por codificar 50 quadros com o QP sendo variado entre 12 a 36.

A observação das Figuras 5.7 a 5.10 nos indica que a taxa de erros na predição dos modos dominantes do próximo quadro sofre menor influência em relação à variação do tamanho do subconjunto de macroblocos empregado na predição mas continua assumindo valores menores para QPs grandes. Não é possível afirmar que exista uma relação direta entre o aumento dos valores de QP e a diminuição do erro de predição, apesar de podermos esperar uma maior redução no tempo de codificação.

Isso nos leva a crer que, para fins de implementação, é possível relaxar o critério de amostragem

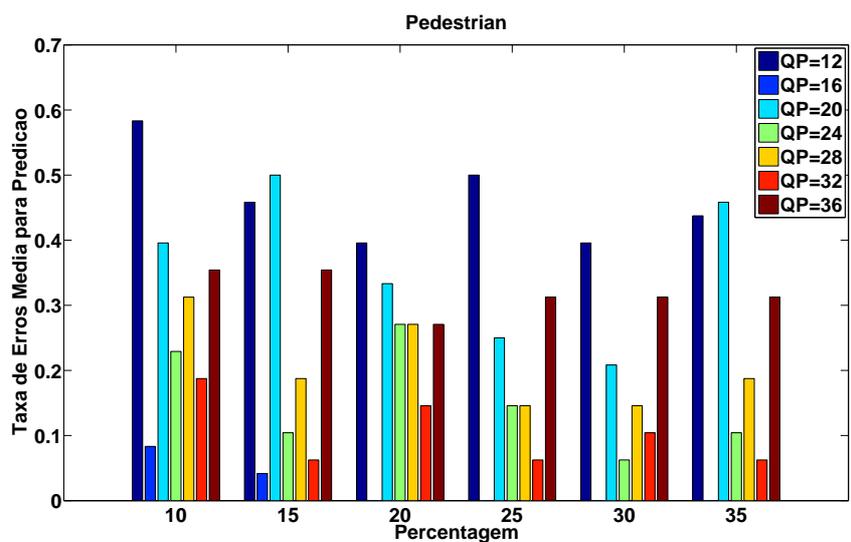


Figura 5.7: Taxa de erro médio na predição dos modos de predição dominantes do próximo quadro  $\times$  porcentagem de macroblocos onde são avaliados todos os modos para a seqüência Pedestrian Area.

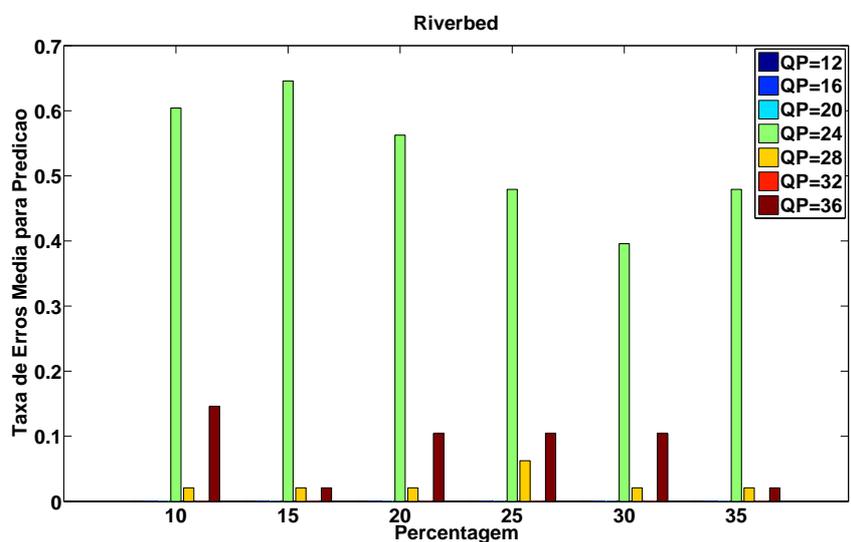


Figura 5.8: Taxa de erro médio na predição dos modos de predição dominantes do próximo quadro  $\times$  porcentagem de macroblocos onde são avaliados todos os modos para a seqüência Riverbed.

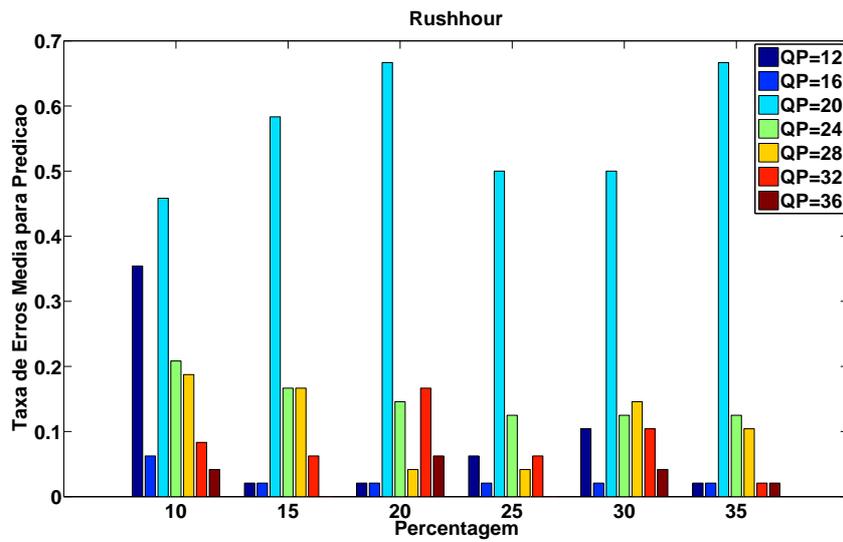


Figura 5.9: Taxa de erro médio na predição dos modos de predição dominantes do próximo quadro  $\times$  porcentagem de macroblocos onde são avaliados todos os modos para a seqüência Rushhour.

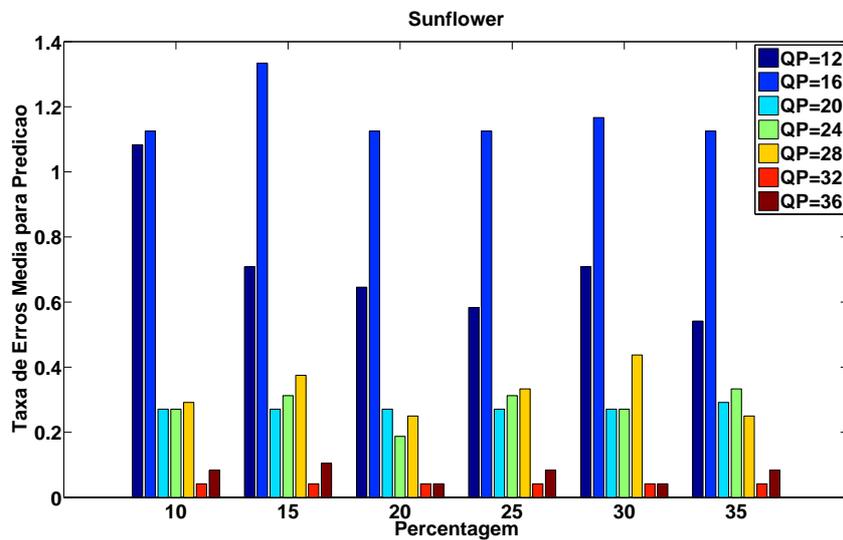


Figura 5.10: Taxa de erro médio na predição dos modos de predição dominantes do próximo quadro  $\times$  porcentagem de macroblocos onde são avaliados todos os modos para a seqüência Sunflower.

apresentado na Seção 5.1.1 para a realização da predição dos modos dominantes do próximo quadro, reduzindo, portanto, o conjunto amostral para 10% dos macroblocos de um quadro de seqüências HD de  $1920 \times 1080$  *pixels* sem impactar muito na qualidade do vídeo entretanto evitando testar os restantes 14,46% dos macroblocos recomendados pela teoria de Estatística Amostrável. [45]

#### 5.1.4 Redutor de Complexidade

Com base nas estatística de ocorrência dos modos de predição do último quadro, propõe-se um **reductor de complexidade** que limitará o uso dos testes de predição. Seu princípio de funcionamento é descrito abaixo.

Codifica-se o primeiro quadro da seqüência como quadro I e o segundo quadro como P da maneira usual em que todos os modos de predição são testados. Para o quadro P, são levantadas as estatísticas de ocorrência dos modos de predição e são elencados os modos dominantes. Para os quadros P seguintes:

1. Em 10% dos macroblocos, escolhidos aleatoriamente, realiza-se o teste completo de predição.
2. Nos 90% dos macroblocos restantes, testam-se apenas os modos dominantes encontrados no quadro P anterior.

O processo acima descrito continua até que apareça novamente um quadro I na seqüência de codificação, o que reinicializa o **reductor**.

## 5.2 RESULTADOS

A metodologia proposta foi implementada no JM-10.2 [43]. As seqüências de teste usadas foram as mesmas empregadas na seção 4.2, que foram: Pedestrian Area, Sunflower, Rush-hour e Riverbed.

De cada seqüência de vídeo foram usados os primeiros 20 quadros e curvas de desempenho taxa  $\times$  distorção foram traçadas por meio da variação do parâmetro de quantização (QP) no intervalo de 8 a 36. Nesse intervalo de QP são atingidos critérios de qualidade e de taxa de transmissão para vídeo de alta definição. Nas Figuras 5.11 a 5.14 são apresentados os resultados das simulações comparando o desempenho do codificador original (JM10.2) e o desempenho do codificador modificado (JM10.2-ANA). A diferença média de PSNR entre as duas curvas de taxa  $\times$  distorção foi calculada de acordo com a metodologia proposta em [44].

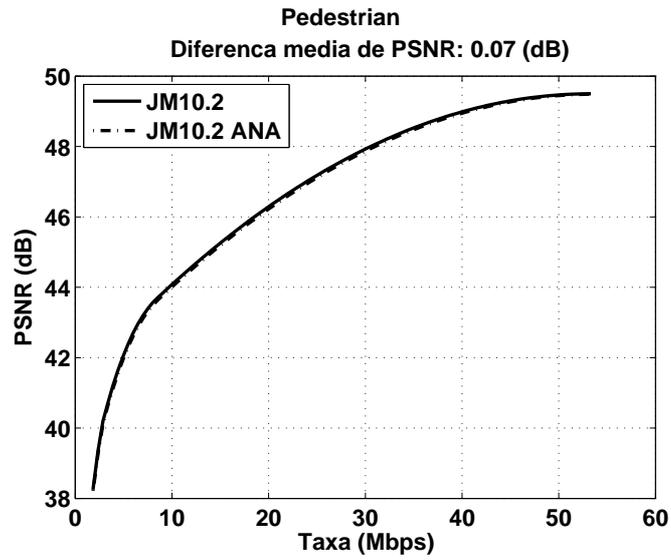


Figura 5.11: Curva de taxa  $\times$  distorção para a seqüência Pedestrian Area: JM10.2  $\times$  JM10.2-ANA.

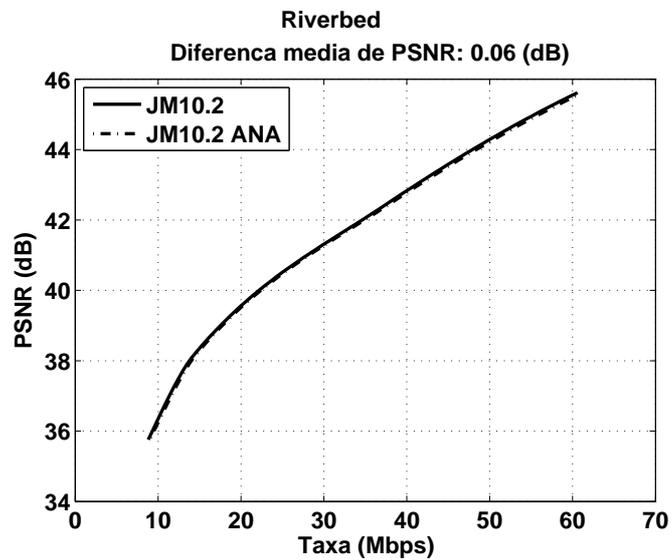


Figura 5.12: Curva de taxa  $\times$  distorção para a seqüência Riverbed: JM10.2  $\times$  JM10.2-ANA.

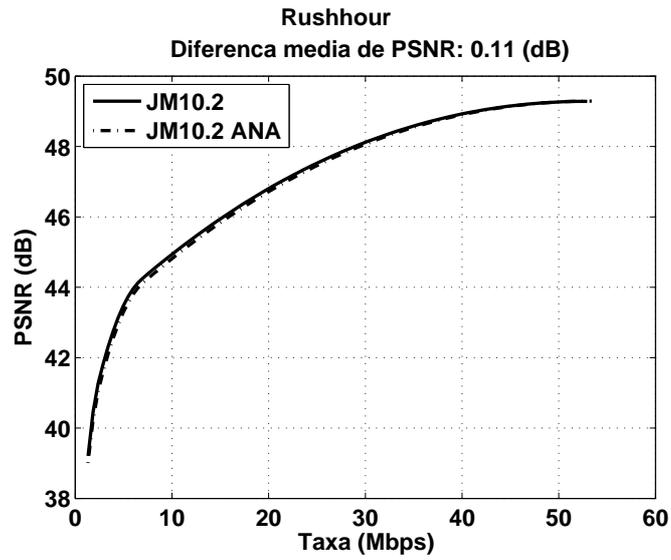


Figura 5.13: Curva de taxa  $\times$  distorção para a seqüência Rushhour: JM10.2  $\times$  JM10.2-ANA.

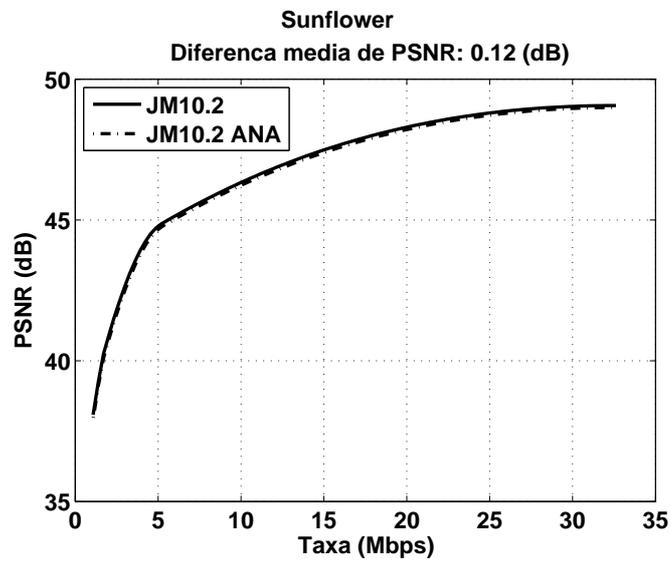


Figura 5.14: Curva de taxa  $\times$  distorção para a seqüência Sunflower: JM10.2  $\times$  JM10.2-ANA.

O perfil de redução de complexidade de codificação é apresentado na Figura 5.15, tomando como referência o tempo requerido na codificação completa.

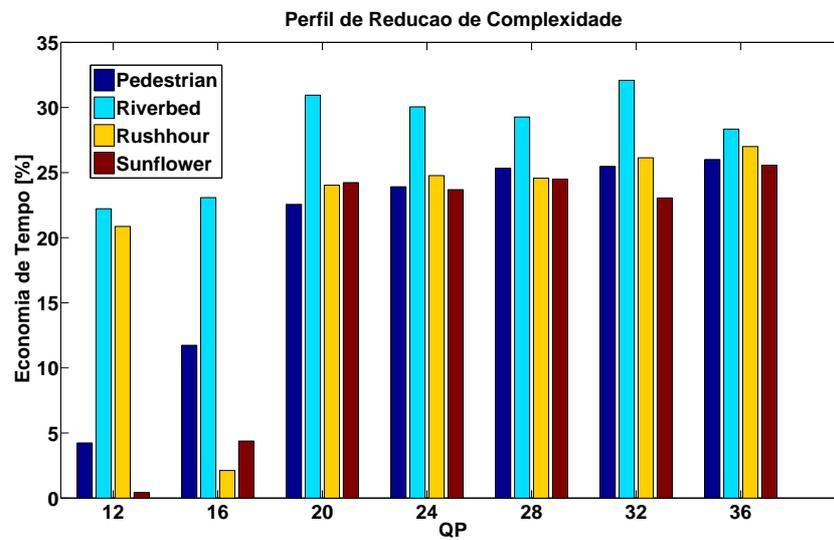


Figura 5.15: Curva do perfil de redução de complexidade (quanto tempo foi economizado) para as quatro seqüências de vídeo analisadas.

Resultados empregando a mesma metodologia para a codificação de seqüências CIF são apresentados nas Figuras 5.16 a 5.18.

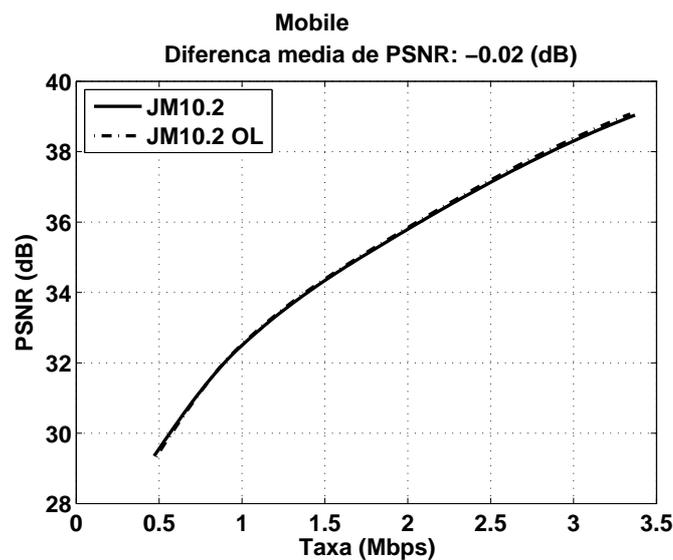


Figura 5.16: Curva de taxa × distorção para a seqüência Mobile: JM10.2 × JM10.2-ANA.

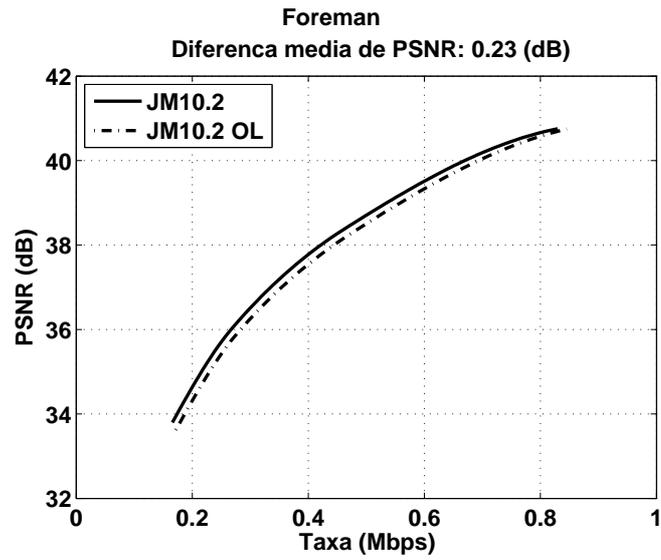


Figura 5.17: Curva de taxa  $\times$  distorção para a seqüência Foreman: JM10.2  $\times$  JM10.2-ANA.

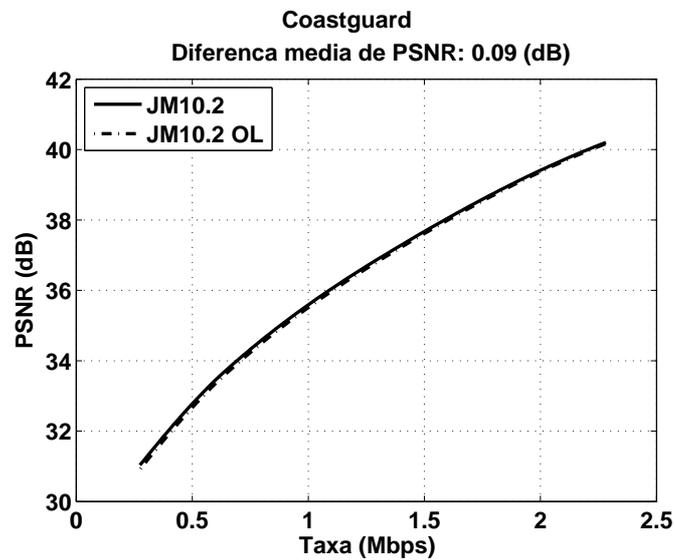


Figura 5.18: Curva de taxa  $\times$  distorção para a seqüência Coastguard: JM10.2  $\times$  JM10.2-ANA.

### 5.3 DISCUSSÃO

As curvas apresentam perdas de qualidade inferiores a 0,15dB quando são usados somente modos dominantes no processo de codificação de uma seqüência de vídeo por meio do H.264/AVC. A diferença é justificada pelos eventuais descasamentos entre o melhor modo de predição encontrado por cada uma das metodologias.

Para valores crescentes de QP, as estatísticas dos modos de predição empregados tendem a ficar polarizadas em um subconjunto bem determinado, ou seja, um número pequeno de modos é responsável por mais de 80% das escolhas de modos de predição; nessa faixa a metodologia é aplicada com sucesso e verifica-se reduções superiores a 20% na complexidade de codificação.

A seqüência Riverbed, cujo conteúdo não é bem predito por meio de estimação de movimentos, tem a peculiaridade de concentrar seus modos de predição em um pequeno subconjunto, primordialmente ao redor de predições Intra-quadro, o que justificou o pequeno descasamento entre a curva de taxa  $\times$  distorção da versão original e da versão modificada com o analisador estatístico. Contudo, à primeira vista, esperaria-se uma maior economia em termos de complexidade computacional de codificação (Figura 5.15). Isso não ocorre devido ao fato de estarmos empregando compensação de movimentos mais inteligentes de forma que, durante a predição, são feitas estimações de movimentos em blocos de  $4 \times 4$  *pixels* e os valores de resíduo para cada vetor de movimento são armazenados e posteriormente usados para a composição de fluxo óptico do quadro para blocos de tamanhos maiores. Portanto, durante a determinação de modos dominantes, se algum dos modos dominantes exigir estimação de movimentos, ela será calculada apenas uma vez; na codificação da seqüência Riverbed, às vezes faz-se necessário adicionar modos que exigem estimação de movimentos, o que a faz ter, em média, a melhor economia em termos de complexidade computacional ao se considerar todas as quadro seqüências, mas não a economia esperada se empregasse somente as técnicas de predição Intra-quadro.

Para seqüências CIF, os resultados de desempenho taxa  $\times$  distorção das Figuras 5.16 a 5.18 mostram que a metodologia provoca poucas perdas de qualidade; contudo, dado que a codificação de seqüências CIF exigem uma complexidade computacional menor, os resultados da aplicação da rotina de análise estatística impactam menos no tempo de codificação da seqüência.



## 6 CONCLUSÕES

Neste trabalho foram propostos métodos de redução de complexidade para o cenário de compressão de vídeos de alta definição com o codificador H.264/AVC.

A primeira contribuição emprega macroblocos do sinal original para a etapa de predição, em vez de usar macroblocos previamente decodificados, o que promove a abertura do laço de predição. O sinal original é usado apenas na realização dos testes de decisão do melhor modo de predição; o resíduo continua sendo calculado a partir dos dados localmente decodificados, descartando a ocorrência de deriva (*drifting*). Testes que consideram o tratamento de vídeos de alta definição pelo codificador H.264/AVC mostram que a perda no desempenho provocada pela nova metodologia é desprezível, tornando atrativa a possibilidade de paralelismo disponibilizada pela técnica.

Uma segunda contribuição, baseada no comportamento estatístico das seqüências, disponibiliza uma maneira menos complexa para realizar os testes de predição. As perdas de desempenho introduzidas pela heurística são pequenas e ainda são contrabalanceadas pela redução da ordem de 25% em tempo de codificação, ganho significativo para o cenário de compressão de vídeo HD.

Cabe salientar que as metodologias dos Capítulos 4 e 5 apresentam modificações em aspectos não normativos da recomendação do H.264/AVC [12], o que significa que as técnicas aqui discutidas não exigem modificações nos decodificadores que seguem a recomendação da ITU-T.

Trabalhos futuros podem se concentrar na implementação de uma estrutura paralelizada para a codificação de sinais de vídeo HD e a agregação da heurística de testes de predição apresentada no Capítulo 5 às estimações de movimento subótimas existentes.

Análises do Capítulo 3, motivações deste trabalho, sugerem que a realização de estimação de movimentos com blocos maiores que os presentes atualmente no H.264/AVC possa ser uma alternativa interessante para a codificação de seqüências de vídeo HD, podendo proporcionar maiores ganhos de codificação.



# REFERÊNCIAS BIBLIOGRÁFICAS

- [1] HUNG, E. M. *Compensação de Movimento utilizando Blocos Multi-escala e Forma Variável em um CODEC de Vídeo Híbrido*. [S.l.]: Dissertação de Mestrado em Engenharia Elétrica com ênfase em Telecomunicações, Publicação PPGENE.DM 304/07, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 73p., 2007.
- [2] RICHARDSON, I. E. G. *H.264 and MPEG-4 Video Compression*. [S.l.]: John Wiley & Sons Ltd, 2003.
- [3] OPPENHEIM, A. V.; SCHAFER, R. W.; BUCK, J. R. *Discrete-Time Signal Processing*. [S.l.]: Prentice Hall, 1998.
- [4] SAYOOD, K. *Introduction to Data Compression*. [S.l.]: Morgan Kuffmann Publishers, 2000.
- [5] ITU-T. *Video codec for audiovisual services at p x 64 kbit/s*. [S.l.], November 1990.
- [6] ISO-IEC JTC1. *Coding of Moving Pictures and Associated Audio for Storage Media at up to About 1.5Mbps - Part 2: Visual*. [S.l.], 1993.
- [7] ITU-T and ISO/IEC JTC 1 - ISO/IEC 13818-2 (MPEG-2). *Generic coding of moving pictures and associating audio information - Part 2: Video*. [S.l.], November 1994.
- [8] HASKEL, B. G.; PURI, A.; NETRAVALLI, A. N. *Digital Video: An Introduction to MPEG-2*. [S.l.]: Chapman and Hall, 1997.
- [9] ITU-T. *ITU-T Recommendation H.263, Video coding for low bit rate communication*. [S.l.], November 2000.
- [10] JTC1, I.-I. *ISO/IEC 14496-2 (MPEG-4 visual version 1), Coding of Audio Visual Objects - Part 2: Visual*. [S.l.], November 1999.
- [11] WIEGAND, T. et al. Rate-Constrained Coder Control and Comparison of Video Coding Standards. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 13, n. 7, p. 688–703, July 2003.
- [12] JVT of ISO/IEC MPEG and ITU-T VCEG. *Advanced Video Coding for Generic Audiovisual Services*. [S.l.], March 2005.

- [13] OSTERMANN, J. et al. Video coding with H.264/AVC: Tools, Performance, and Complexity. *IEEE Circuits and Systems Magazine*, p. 7–28, 2004.
- [14] HUANG, Y.-Y. et al. Analysis and complexity reduction of multiple reference frames motion estimation in H.264/AVC. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 16, n. 4, p. 507–522, April 2006.
- [15] HE, Z. et al. Power-Rate-Distortion Analysis for Wireless Video Communication Under Energy Constraints. *IEEE Transactions on Circuits and Systems for Video Technology*, p. 645–658, May 2005.
- [16] ALLEBACH, J. P. Image scanning, sampling and interpolation. In: BOVIK, A. (Ed.). *Handbook of Image and Video Processing*. [S.l.]: Academic Press, 2000. p. 629–644.
- [17] DUBOIS, E. Video sampling and interpolation. In: BOVIK, A. (Ed.). *Handbook of Image and Video Processing*. [S.l.]: Academic Press, 2000. p. 645–654.
- [18] DINIZ, P. S. R.; SILVA, E. A. B. da; NETTO, S. L. *Digital Signal Processing*. [S.l.]: Cambridge University Press, 2006. (Series in Imaging Science and Technology: System Analysis and Design).
- [19] SHARMA, G. Color fundamentals for digital imaging. In: SHARMA, G. (Ed.). *Digital Color Imaging Handbook*. [S.l.]: CRC Press, 2003. p. 1–114.
- [20] HUNT, R. W. G. *The Reproduction of Colour*. England: John Wiley and Sons, 2004. (Series in Imaging Science and Technology).
- [21] Recommendation ITU-T BT.601-5. *Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios*. [S.l.], 1995.
- [22] PENNEBAKER, W. B.; MITCHELL, J. L. *JPEG: Still Image Data Compression Standard*. New York, USA: Van Nostrand Reinhold, 1993.
- [23] PRATT, W. K. *Digital Image Processing: PIKS Inside*. California, USA: Wiley-Interscience, 2001.
- [24] WU, H. R.; RAO, K. R. Critical issues and challenges. In: WU, H. R.; RAO, K. R. (Ed.). *Digital Video Image Quality and Perceptual Coding*. [S.l.]: CRC Press, 2006. p. 543–573.
- [25] TAUBMAN, D. S.; MARCELLIN, M. W. *JPEG2000: Image Compression Fundamentals, Standards and Practice*. [S.l.]: Kluwer Academic, 2002.
- [26] IMAGE Coding System: Motion JPEG 2000 (JPEG2000 Part 3). [S.l.], September 2003.

- [27] QUEIROZ, R. L. de et al. Fringe Benefits of the H.264/AVC. In: *International Telecommunication Symposium*. [S.l.: s.n.], 2006. p. 208–212.
- [28] SULLIVAN, G. J.; TOPIWALA, P.; LUTHRA, A. The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions. *Proc. SPIE Conference on Applications of Digital Image Processing XXVII, Special Session on Advances in the New Emerging Standard: H.264/AVC*, August 2004.
- [29] MALVAR, H. S. et al. Low-Complexity transform and quantization in H.264/AVC. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 13, n. 13, p. 590–603, July 2003.
- [30] MARPE, D.; SCHWARZ, H.; WIEGAND, T. Context-based adaptative binary arithmetic coding in H.264/AVC video compression standard. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 13, p. 620–636, July 2003.
- [31] LIST, P. et al. Adaptive deblocking filter. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 13, p. 614–619, July 2003.
- [32] COTE, G. et al. H.263+: Video coding at low bit rates. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 8, p. 849–866, November 1998.
- [33] LUTHRA, A.; SULLIVAN, G. J.; WIEGAND, T. H.264/AVC Video Coding Standard. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 13, n. 7, July 2003.
- [34] ITU-T. *Amendment 1: Support of additional colour spaces and removal of the High 4:4:4 Profile*. [S.l.], June 2006.
- [35] TOURAPIS, H.-Y. C.; TOURAPIS, A. M.; TOPIWALA, P. Fast Motion Estimation within the JVT Codec. *JVT-E023, 5th Meeting: Geneva, Switzerland*, October 2002.
- [36] CHEN, Z.; ZHOU, P.; HE, Y. Hybrid Unsymmetrical-cross Multi-Hexagon-grid Search Strategy for Integer Pel Motion Estimation in H.264. *Picture Coding Symposium*, April 2003.
- [37] TOURAPIS, H.-Y. C.; TOURAPIS, A. M.; TOPIWALA, P. Fast Motion Estimation within the H.264 Codec. *Proc. of International Conference on Multimedia and Expo. ICME*, v. 3, p. 517–520, July 2003.
- [38] KUO, T.-Y.; CHAN, C.-H. Fast Variable Block Size Motion Estimation for H.264 Using Likelihood and Correlation of Motion Field. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 16, n. 10, p. 1185–1195, October 2006.

- [39] KIM, B.; SONG, S.-K.; CHO, C.-S. Efficient Inter-Mode Decision Based on Contextual Prediction for the P-Slice in H.264/AVC Video Coding. *IEEE International Conference on Image Processing*, p. 1333–1336, September 2006.
- [40] LA, B.; EOM, M.; CHOE, Y. Fast Mode Decision for Intra Prediction in H.264/AVC Encoder. *IEEE International Conference on Image Processing*, V, p. 321–324, September 2007.
- [41] HWANG, C.; ZHUANG, S.; LAI, S.-H. Efficient Intra Mode Selection Using Image Structure Tensor for H.264/AVC. *IEEE International Conference on Image Processing*, V, p. 289–292, September 2007.
- [42] AKYOL, E.; MUKHERJEE, D.; LIU, Y. Complexity Control for Real-Time Video Coding. *IEEE International Conference on Image Processing*, I, p. 77–80, September 2007.
- [43] JVT. JM 10.2. <http://bs.hhi.de/~suehring>.
- [44] BJONTEGAARD, G. Calculation of average PSNR differences between RD-curves. *Coc. VCEG-M33*, April 2001.
- [45] FONSECA, J. S. da; MARTINS, G. de A. *Curso de Estatística*. [S.l.]: Editora Atlas, 1996.
- [46] COCHRAN, W. G. *Sampling Techniques*. [S.l.]: John Wiley & Sons Ltd, 1997.