

UNIVERSIDADE FEDERAL DE PELOTAS
Centro de Desenvolvimento Tecnológico
Programa de Pós-Graduação em Computação

Dissertação



**Projeto e Gerenciamento Dinâmico de uma Memória de Busca para a
Estimação de Movimento na Codificação de Vídeo**

Lívia Silva do Amaral

Pelotas, 2018

Lívia Silva do Amaral

**Projeto e Gerenciamento Dinâmico de uma Memória de Busca para a
Estimação de Movimento na Codificação de Vídeo**

Dissertação apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Mestre em Ciência da Computação

Orientador: Prof. Dr. Marcelo Schiavon Porto
Coorientadores: Prof. Dr. Bruno Zatt
Prof. Dr. Luciano Volcan Agostini

Pelotas, 2018

Universidade Federal de Pelotas / Sistema de Bibliotecas
Catalogação na Publicação

A111p Amaral, Livia Silva do

Projeto e gerenciamento dinâmico de uma memória de busca para a estimação de movimento na codificação de vídeo / Livia Silva do Amaral ; Marcelo Schiavon Porto, orientador ; Bruno Zatt, Luciano Volcan Agostini, coorientadores. — Pelotas, 2018.

120 f. : il.

Dissertação (Mestrado) — Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, 2018.

1. Codificação de vídeo. 2. Redução de consumo de energia. 3. HEVC. 4. Controle dinâmico de ME. 5. Test zone search. I. Porto, Marcelo Schiavon, orient. II. Zatt, Bruno, coorient. III. Agostini, Luciano Volcan, coorient. IV. Título.

CDD : 005

RESUMO

AMARAL, Livia Silva do. **Projeto e Gerenciamento Dinâmico de uma Memória de Busca para a Estimação de Movimento na Codificação de Vídeo**. 2018. 120f. Trabalho Individual (Mestrado em Ciência da Computação) – Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2018.

O número elevado de acessos à memória durante a codificação de vídeo gera uma preocupação grande com o consumo de energia e a vida útil da bateria de dispositivos que lidam com vídeos digitais. Na tentativa de reduzir a redundância temporal presente em quadros vizinhos, a estimação de movimento (*Motion Estimation* - ME) busca blocos de quadros de referência na memória externa e compara ao bloco atual, que está sendo codificado. Esta comunicação entre a unidade de processamento e a memória é numerosa e gera um elevado consumo de energia, resultando em uma menor durabilidade de bateria em dispositivos móveis. Neste sentido, este trabalho realiza uma análise do comportamento dos acessos do algoritmo de estimação de movimento *Test Zone Search* e, a partir dela, propõe um projeto de memória de busca e um gerenciador dinâmico de estimação de movimento centrado nesta memória. Este gerenciador divide a memória interna em dois setores e desliga o setor mais externo quando este não é relevante para a ME. Com o desligamento de setores, este gerenciamento dinâmico reduz o consumo estático e também diminui o número de acessos durante a ME, reduzindo também o consumo dinâmico. Neste trabalho, onze modos de gerenciamento dinâmico foram avaliados. Com um desses modos e para um conjunto variado de vídeos de alta resolução, o uso do gerenciamento dinâmico de memória alcançou uma média de 43,59% de redução no consumo de energia, com uma variação negligível em BD-Rate de 0,49%.

Palavras-chave: codificação de vídeo; HEVC; *test zone search*; redução de consumo de energia; controle dinâmico de ME

ABSTRACT

AMARAL, Livia Silva do. **Design and Dynamic Management of a Search Memory for Motion Estimation in Video Coding**. 2018. 120f. Trabalho Individual (Mestrado em Ciência da Computação) – Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2018.

The high number of memory accesses during video encoding leads to a high concern related to energy consumption and battery life of mobile devices that handle digital videos. In an attempt to reduce the temporal redundancy present in neighboring frames, the ME searches for blocks of reference frames in the external memory and compares it to the current block, which is being coded. This communication between the processing unit and the external memory is numerous and causes a high energy consumption, resulting in a shorter battery life in mobile devices. Thus, considering the high number of accesses to external memory, solutions for the reduction of memory bandwidth in video coding systems should be explored. In this sense, this work performs an analysis of the distribution of the accesses of the algorithm of motion estimation Test Zone Search and, from this analysis, proposes a static memory design and a dynamic management for motion estimation from this memory. The dynamic manager splits the internal memory into two sectors and shuts off the outermost sector when it is not relevant to ME. With the sectors shutdown, this dynamic management reduces the static consumption and also decreases the number of accesses during the ME, also reducing the dynamic consumption. In this work, eleven modes of dynamic management were evaluated. With one of these modes and for a varied set of high-resolution videos, the use of memory dynamic management achieved an average of 43.59% reduction in power consumption, with negligible variation in BD-Rate of 0.49%.

Keywords: video coding; HEVC; test zone search; energy consumption reduction; ME dynamic control

Lista de Figuras

Figura 1. Diagrama de blocos simplificado de um codificador de vídeo	18
Figura 2. Particionamento de uma CTU: (a) exemplo de particionamento de uma CTU 64x64 e (b) descrição deste particionamento em forma de árvore quadrática..	19
Figura 3. Representação da estimação de movimento: (a) quadro de referência e (b) quadro atual (AMARAL, 2015)	22
Figura 4. Fluxograma do algoritmo de busca TZS	23
Figura 5. Ilustração da etapa de Pré-Busca e seus preditores.....	24
Figura 6. Busca Inicial do TZS em formato losangular.....	25
Figura 7. Ilustração da Busca Raster, com $iRaster = 2$	27
Figura 8. Ilustração da fase de Refinamento: (a) primeira iteração e (b) segunda iteração	28
Figura 9. Estrutura do <i>trace</i> utilizado.....	38
Figura 10. Comportamento dos acessos na área de busca de cada vídeo.....	41
Figura 11. Comportamento médio dos acessos durante a codificação dos vídeos analisados: (a) visão lateral e (b) visão superior	43
Figura 12. Relação entre os dados mais acessados e a área onde se encontram ...	44
Figura 13. Análise dos acessos para codificar uma linha de CTUs do vídeo <i>BasketballDrive</i> : (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo	46
Figura 14. Análise dos acessos para codificar uma linha de CTUs do vídeo <i>Kimono</i> : (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo.....	46
Figura 15. Análise dos acessos para codificar uma linha de CTUs do vídeo <i>Tennis</i> : (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo.....	47
Figura 16. Análise dos acessos para codificar uma linha de CTUs do vídeo <i>BQTerrace</i> : (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo.....	47
Figura 17. Análise dos acessos para codificar uma linha de CTUs do vídeo <i>Cactus</i> : (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo.....	48
Figura 18. Análise dos acessos para codificar uma linha de CTUs do vídeo <i>ParkScene</i> : (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo.....	48
Figura 19. Sistema avaliado neste capítulo.....	50
Figura 20. <i>Bitmap</i> referente ao corte de 33,35% da área de busca	51
Figura 21. Relação entre as áreas de busca: (a) <i>search range</i> [-64, +63] e	53
Figura 22. <i>Bitmaps</i> referentes aos cortes fixos: (a) corte fixo de 65,60% e (b) corte fixo de 81,36% da área de busca.....	55
Figura 23. Divisão da área de busca em quatro setores	62
Figura 24. Mapa de setores adotado.....	65
Figura 25. Organização dos dados no <i>Level C</i> (AMARAL, 2015)	66
Figura 26. Comportamento das células SRAM no caso O-O-O: (a) área de busca da CTU A, (b) área de busca da CTU B e (c) área de busca da CTU C	67
Figura 27. Matriz de referência gerada pela codificação do Quadro 1 do vídeo <i>BQTerrace</i>	70
Figura 28. Divisão do quadro em quadrantes.....	70

Figura 29. Setores ativos na codificação do Quadro 2 do vídeo <i>BQTerrace</i> com a solução GD 4Q 1/3.....	71
Figura 30. Setores ativos na codificação do Quadro 2 do vídeo <i>BQTerrace</i> com a solução GD L 1/3.....	72
Figura 31. Setores ativos na codificação do Quadro 2 do vídeo <i>BQTerrace</i> com a solução GD ML 1/3.....	73
Figura 32. Ocorrência de overhead (caso <i>O-I-O</i>): (a) área de busca da CTU A, (b) área de busca da CTU B e (c) área de busca da CTU C	74
Figura 33. Solução otimizada para evitar o overhead no caso <i>O-I-O</i> : (a) área de busca da CTU A, (b) área de busca da CTU B e (c) área de busca da CTU C.....	75
Figura 34. Setores ativos na codificação do Quadro 2 do vídeo <i>BQTerrace</i> com gerenciamento dinâmico: (a) solução GD CTU e (b) solução GD CTU OT.....	76
Figura 35. Ilustração da solução GD MV: (a) matriz de referência e (b) setores ativados.....	77
Figura 36. Ilustração da solução GD ESO: (a) matriz de referência e (b) setores ativados.....	78
Figura 37. Ilustração da solução GD <i>Blur</i> : (a) matriz de referência e (b) setores ativados.....	79
Figura 38. Ilustração da solução GD JCTU: (a) matriz de referência e (b) setores ativados.....	79
Figura 39. Gerenciamentos dinâmicos para codificar o Quadro 2 do vídeo <i>BQTerrace</i>	80
Figura 40. Resultados de BD-Rate.....	84
Figura 41. Resultados entre os consumos estático e dinâmico da SPM.....	85
Figura 42. Resultados de redução de consumo das soluções de GD.....	86
Figura 43. Relação entre os resultados de redução do consumo e BD-Rate.....	89
Figura 44. Resultados finais de redução de consumo das soluções de GD	93
Figura 45. Relações entre os consumos estático e dinâmico da SPM dos resultados finais.....	94
Figura 46. Relações entre os resultados finais de redução de consumo e BD-Rate.....	95

Lista de Tabelas

Tabela 1. Resultados de <i>bitrate</i> (bits/s) e PSNR (dB) para search range 64, 32 e corte fixo de 33,35%.....	54
Tabela 2. Resultados de <i>bitrate</i> (bits/s) e PSNR (dB) para cortes fixos de 65,60% e 81,36% em relação à área de busca original	56
Tabela 3. Resultados de BD-Rate para as variações de tamanho da área de busca	58
Tabela 4. Total de amostras acessadas pelos tamanhos de áreas de busca	59
Tabela 5. Redução do total de amostras acessadas.....	59
Tabela 6. Média dos acessos por setor.....	63
Tabela 7. Ocorrência dos setores mais relevantes em número de CTUs (%).....	64
Tabela 8. Informações gerais das SPMs.....	83
Tabela 9. Resultados de BD-Rate e redução de consumo das soluções retornadas pela frente de Pareto.....	90
Tabela 10. Resultados de BD-Rate expandidos.....	91
Tabela 11. Resultado de consumo de energia total da SR [-64, +63]* e reduções de consumo de energia (%) obtidos pelo uso do <i>Level C</i> , DDRFC e CF ou GD.....	98

LISTA DE ABREVIATURAS E SIGLAS

AMSR	<i>Adaptive Motion Search Range</i>
ASRME	<i>Adaptive Search Range Motion Estimation</i>
BD-PSNR	<i>Bjontegaard Delta PSNR</i>
BD-Rate	<i>Bjontegaard Delta Rate</i>
CABAC	<i>Context-Adaptive Binary Arithmetic Coding</i>
CF	<i>Corte Fixo</i>
CIF	<i>Common Intermediate Format</i>
CTU	<i>Coding Tree Unit</i>
CU	<i>Coding Unit</i>
DCT	<i>Discrete Cosine Transform</i>
DDRFC	<i>Double Differential Reference Frame Compressor</i>
DPB	<i>Decoded Picture Buffer</i>
DRAM	<i>Dynamic Random-Access Memory</i>
DRFVLC	<i>Differential Reference Frame Variable-Length Compressor</i>
DSR	<i>Dynamic Search Range</i>
EC	<i>Consumo de Energia</i>
ESO	<i>Expansão do Setor O</i>
FS	<i>Full Search</i>
GD	<i>Gerenciamento Dinâmico</i>
HD	<i>High Definition</i>
HEVC	<i>High Efficient Video Coding</i>
HM	<i>HEVC Test Model</i>
JCTU	<i>Janela de CTUs</i>
k-NN	<i>k-Nearest Neighbor</i>
LPDDR	<i>Low Power Double Data Rate</i>
ME	<i>Motion Estimation</i>
MSE	<i>Mean Square Error</i>
MV	<i>Maioria dos Vizinhos</i>

PC	Preditor bloco colocalizado
PE	Preditor bloco imediatamente à esquerda
PM	Preditor mediana
PS	Preditor bloco imediatamente superior
PSD	Preditor bloco imediatamente superior direito
PSNR	<i>Peak Signal-to-Noise Ratio</i>
PU	<i>Processing Unit</i>
QCIF	<i>Quarter Common Intermediate Format</i>
QP	<i>Quantization Parameter</i>
SAD	<i>Sum of Absolute Differences</i>
SBTVD	Sistema Brasileiro de Televisão Digital
SDRAM	<i>Synchronous Dynamic Random Access Memory</i>
SPM	<i>Scratchpad memory</i>
SR	<i>Search Range</i>
SRAM	<i>Static Random Access Memory</i>
TZS	<i>Test Zone Search</i>
UHD	<i>Ultra High Definition</i>

SUMÁRIO

1	INTRODUÇÃO	13
2	CODIFICAÇÃO DE VÍDEO	16
2.1	Estimação de movimento e acessos à memória.....	20
2.2	O algoritmo Test Zone Search.....	23
2.2.1	Pré-Busca.....	24
2.2.2	Busca Inicial	24
2.2.3	Busca <i>Raster</i>	26
2.2.4	Refinamento	27
2.3	Considerações finais do capítulo.....	28
3	CONSUMO DE ENERGIA NA ESTIMAÇÃO DE MOVIMENTO E ANÁLISE DOS ACESSOS À MEMÓRIA	30
3.1	Trabalhos relacionados	32
3.1.1	Reuso de dados.....	32
3.1.2	Algoritmos para DSR	33
3.2	Análise da distribuição dos acessos	37
3.3	Análise dos acessos em regiões do mesmo quadro.....	44
3.4	Considerações finais do capítulo.....	49
4	ESTIMAÇÃO DE MOVIMENTO POR MEIO DE CORTES FIXOS NA ÁREA DE BUSCA.....	50
4.1	Projeto da memória de busca.....	61
5	GERENCIAMENTO DINÂMICO DA ESTIMAÇÃO DE MOVIMENTO	69
5.1	Gerenciamento por quadrantes (GD 4Q 1/3 e GD 4Q 1/5).....	70
5.2	Gerenciamento por linha (GD L 1/2 e GD L 1/3).....	71
5.3	Gerenciamento por meia linha (GD ML 1/3)	72
5.4	Gerenciamento por CTU (GD CTU e GD CTU OT).....	72
5.5	Gerenciamento pela maioria dos vizinhos (GD MV).....	77
5.6	Gerenciamento pela expansão do Setor O (GD ESO)	78
5.7	Gerenciamento por meio do <i>Blur</i> (GD <i>Blur</i>)	78
5.8	Gerenciamento pela janela de CTUs (GD JCTU).....	79

5.9	Modelo para avaliação de consumo de energia.....	81
5.10	Resultados	83
6	RESULTADOS FINAIS E DISCUSSÕES	89
6.1	Redução de consumo do sistema avaliado	96
7	CONCLUSÕES	100
	APÊNDICE A	104
	APÊNDICE B	105
	APÊNDICE C	111
	REFERÊNCIAS.....	117

1 INTRODUÇÃO

Recentes avanços ao longo dos anos vêm proporcionando uma rápida evolução tecnológica que se reflete também no mercado, onde *gadgets* têm vida curta. Dispositivos móveis, como *smartphones* e *tablets*, são lançados e logo superados por novos aparelhos cada vez mais potentes e capazes de realizar um maior número de tarefas. Esta revolução tem permitido que os aparelhos apresentem preços convidativos aliados a uma maior qualidade de serviços de mídia, como qualidade de áudio/vídeo e capacidade de gravar e transmitir conteúdos multimídia, entre outros. A grande variedade de opções de modelos e preços vem tornando estes dispositivos acessíveis e populares entre todos os níveis sociais. A tendência, segundo um relatório publicado pela CISCO, é que o tráfego de dados móveis aumentará sete vezes entre 2016 e 2021 (CISCO, 2017).

Desta forma, vídeos digitais também têm se popularizado. Dados relacionados a vídeos ocuparam 73% do total de tráfego de dados na Internet em 2016 e o total de tráfego relacionado a vídeos crescerá quatro vezes até 2021 (CISCO, 2017). Além disso, até 2021, a cada segundo, um milhão de minutos de conteúdo de vídeo cruzará a Internet (CISCO, 2017). Esse volume de dados é impulsionado pelo consumo e compartilhamento de vídeos em redes sociais. Em 2016, o tempo de exibição diário para transmissões do Facebook Live quadruplicou e houve um aumento de 80% no tempo que as pessoas passaram assistindo vídeos no Instagram (FACEBOOK, 2017). Além disso, segundo relatórios do Instagram, o número de vídeos postados por dia na rede social vem crescendo quatro vezes a cada ano (INSTAGRAM, 2017).

Para satisfazer usuários e para que eles possam capturar e armazenar os vídeos que pretendem compartilhar e transmitir, os dispositivos móveis que suportam vídeos digitais devem ser capazes de lidar com vídeos com qualidade e resoluções cada vez maiores. Esta necessidade, porém, implica na quantidade de dados que precisam ser armazenados na memória destes aparelhos para que o armazenamento destes vídeos se torne possível. Aliado a isto, está o fato de que a largura de banda necessária entre a memória e o codificador cresce conforme o tamanho do vídeo e o número de vistas, no caso de vídeos 3D. Além disto, grande parte dos dispositivos

móveis é alimentada por baterias. Isto traz restrições e desafios em relação ao consumo de energia e o processamento feito por estes aparelhos. Deste modo, este cenário reforça a necessidade de uma codificação de vídeo eficiente, para que se torne viável a manipulação de conteúdo multimídia por dispositivos móveis.

O atual padrão de codificação de vídeo estado-da-arte é o *High Efficient Video Coding* (HEVC) (ITU-T, 2013), que prevê uma redução de 50% na taxa de bits quando comparado com o seu antecessor, o H.264/AVC (ITU-T, 2012), mantendo a mesma qualidade visual. O padrão H.264/AVC é consolidado no mercado, na literatura e é o atual padrão adotado pelo Sistema Brasileiro de Televisão Digital (SBTVD). Ambos os padrões definem diversas técnicas, que quando utilizadas em conjunto, possibilitam uma alta taxa de compressão com perda de qualidade de vídeo muitas vezes imperceptível aos olhos humanos. Entre as principais ferramentas adotadas pelos padrões atuais de codificação de vídeo está a estimação de movimento (*Motion Estimation* – ME).

A ME é a etapa que mais contribui para as taxas de compressão do codificador (PESQUET-POPESCU, CAGNAZZO e DUFAUX, 2014), mas também é a que mais demanda esforço computacional (GRELLERT, BAMPI e ZATT, 2016). Esta etapa é responsável por encontrar a melhor correspondência entre o bloco do quadro que está sendo codificado e blocos de quadros de referência, que já foram previamente codificados. Esta busca pela melhor correspondência ocorre dentro de uma região denominada área de busca, localizada nos quadros de referência. Um quadro de referência é armazenado na memória externa logo após passar pelo seu processo de codificação. Estes acessos realizados na memória externa, principalmente aos quadros de referência, têm alto consumo de energia e representam um gargalo no sistema como um todo (ZATT, *et al.*, 2011), onde cerca de 70% a 90% do consumo da ME está relacionado com os acessos à memória (SAMPAIO, *et al.*, 2014)

Estes acessos geram um alto consumo de energia, o que não é desejável, especialmente, quando dispositivos alimentados a bateria são considerados. Para amenizar este problema existem soluções já consolidadas na literatura, que se baseiam em três abordagens diferentes: compressores de quadros de referência, reuso de dados e aplicação de um ajuste dinâmico na área de busca.

Compressores de quadros de referência aplicam um conjunto de técnicas para comprimir o quadro antes que ele seja salvo na memória externa e realizar a descompressão quando o mesmo é solicitado pela ME. O reuso de dados envolve o

emprego de memórias rápidas situadas entre a memória externa e o codificador de vídeo e são responsáveis por armazenar dados que são frequentemente buscados pela ME na memória externa, diminuindo o número de acessos feitos nesta memória. Por fim, a aplicação de um ajuste dinâmico na área de busca permite que ela sofra variações em seu tamanho durante a codificação do vídeo, visando reduzir a quantidade de dados lidos da memória durante a ME quando estes são desnecessários, sem causar grandes impactos na eficiência de codificação.

O cenário atual exige que soluções de redução do consumo de energia em codificadores de vídeos digitais sejam empregadas. No caso de dispositivos alimentados a bateria, essas soluções se tornam indispensáveis. Além disso, técnicas com baixo custo de implementação em hardware e baixo impacto na eficiência de codificação são altamente desejáveis. Também é importante ressaltar que as técnicas apresentadas e discutidas anteriormente podem ser utilizadas em conjunto e de forma complementar.

Neste sentido, com foco em possibilitar uma redução do consumo de energia ainda maior do que a já alcançada pela união das técnicas de reuso de dados e compressão de quadros de referência e atingir os menores impactos possíveis em eficiência de codificação, este trabalho propõe um projeto de memória de busca, que permite *power gating* a nível de setores, e um gerenciador dinâmico da estimação de movimento centrado nesta memória. No sistema proposto, a área de busca é armazenada em uma memória interna rápida e dividida em dois setores de memória. O setor mais externo pode ser desligado (ou ligado) conforme o número de requisições feitas na ME. Desligando o setor externo, há uma redução no consumo estático da memória interna. Além disso, menos blocos são acessados durante a ME, reduzindo também o consumo dinâmico do sistema.

Este gerenciamento dinâmico é baseado em um estudo estatístico do comportamento dos acessos do algoritmo *Test Zone Search* dentro de uma área de busca e pode assumir diferentes estratégias de ativação e desativação do setor mais externo observando uma relação entre consumo de energia e qualidade de vídeo. Assim, diferentes técnicas de gerenciamento dinâmico de estimação de movimento foram avaliadas. Os resultados mostraram que o uso do gerenciamento dinâmico pode chegar a uma redução de 43,59% consumo de energia para um conjunto variado de vídeos de alta resolução, com impacto negligível em eficiência de codificação.

2 CODIFICAÇÃO DE VÍDEO

Vídeos digitais são formados por uma sequência de imagens estáticas, ou quadros, que são dispostas a uma determinada frequência de exibição. Cada quadro é composto por pixels, que representam a menor informação que forma uma imagem digital. Vídeos coloridos requerem que cada pixel relacione três canais de informação para que uma determinada cor seja representada com precisão dentro de um espaço de cor (RICHARDSON, 2010). O espaço de cor mais utilizado em vídeos digitais é o YCbCr, onde o Y representa as informações de luminância, Cb, as informações de croma azul e Cr, as informações de croma vermelha. Assim, cada pixel é formado por uma amostra de luminância e duas amostras de croma. Este formato é vantajoso porque, dado que o sistema visual humano é mais sensível as informações de luminância do que de croma, as informações de luz são separadas das informações de cor e, assim, as informações de cor podem ser representadas com um menor número de bits (RICHARDSON, 2010). Por meio desta técnica, uma redução significativa na quantidade de informações de um vídeo pode ser obtida, sem causar perdas significativas na sua qualidade visual.

Ainda assim, vídeos descomprimidos necessitam de uma vasta quantidade de dados para serem representados. Além disto, há uma constante procura por vídeos com resoluções, frequências de exibição e fidelidade visual cada vez maiores (SZE, BUDAGAVI e SULLIVAN, 2014). Estes elementos aumentam ainda mais a quantidade de dados necessários para a representação de um vídeo. Assim, para que a transmissão e o armazenamento destes vídeos se tornem tarefas viáveis, uma codificação de vídeo eficiente se faz necessária. Uma característica importante nos vídeos digitais é que, embora eles necessitem de uma grande quantidade de dados, eles também apresentam elevado grau de redundância (RICHARDSON, 2002). Isto significa que grande parte da enorme quantidade de dados utilizada na representação de um vídeo pode ser descartada, sem perdas significativas na qualidade visual. Desta forma, o principal objetivo do processo de codificação de vídeo se encontra na máxima eliminação das redundâncias presentes nos vídeos.

No escopo da codificação de vídeo 2D existem três tipos de redundância: redundância temporal, redundância espacial e redundância entrópica (RICHARDSON, 2010). A redundância espacial, ou intra-quadro, é percebida por meio da semelhança entre as amostras que compõem o quadro. Da mesma forma, a semelhança de amostras ou blocos entre quadros vizinhos também compõem uma nova redundância. Esta é chamada de redundância temporal, ou inter-quadro, e surge a partir das altas taxas de amostragem apresentadas pelos vídeos digitais que fazem com que os quadros vizinhos de um vídeo sejam muito semelhantes entre si. Assim, uma análise eficiente da redundância temporal pode gerar altas taxas de compressão. Por fim, a redundância entrópica está relacionada com a probabilidade de ocorrência dos símbolos codificados e com uma representação mais eficiente destes símbolos, a fim de transmitir a maior quantidade de informação possível por símbolo. Algoritmos de compressão sem perdas são utilizados nesta etapa.

A qualidade de um vídeo codificado pode ser medida de duas formas: qualidade subjetiva e qualidade objetiva (RICHARDSON, 2002). A primeira diz respeito à impressão de determinado indivíduo sobre o vídeo. Porém, avaliar a eficiência de codificação por meio da qualidade subjetiva se torna muito difícil. Assim, ao invés da qualidade subjetiva, os codificadores de vídeo utilizam métricas de qualidade objetiva. A métrica mais utilizada é o *Peak-to-Signal Noise Ratio* (PSNR). O PSNR é medido em uma escala logarítmica e expresso em unidades decibéis (dB). Além disso, esta métrica é baseada no valor do erro quadrático médio (*Mean Square Error* - MSE) entre o quadro original e o quadro reconstruído. As Equações 1 e 2 mostram as definições do MSE e do PSNR, respectivamente, onde R é uma matriz que representa as amostras do quadro reconstruído e O representa as amostras do quadro original.

$$MSE(x, y) = \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} (R_{i,j} - O_{i,j})^2 \quad (1)$$

$$PSNR = 10 * \log_{10} \left(\frac{MAX^2}{MSE} \right) \quad (2)$$

Na Equação 2, MAX representa o maior valor de uma amostra.

No sentido de promover uma codificação de vídeo eficiente e descartar a redundância presente nos vídeos, surgem os padrões de codificação de vídeo. Os

padrões de codificação de vídeo atuais incluem o padrão H.264/AVC e o padrão HEVC. Enquanto o H.264/AVC é o padrão adotado pelo SBTVD, o HEVC, lançado em 2013, é o atual padrão estado-da-arte. Uma série de novas ferramentas foram introduzidas pelo HEVC, a fim alcançar o dobro da eficiência de codificação atingida pelo H.264/AVC. Os próximos parágrafos serão dedicados a descrever brevemente o processo de codificação de um determinado vídeo, com foco na codificação do padrão HEVC. A Figura 1 ilustra um diagrama de blocos do fluxo de codificação de vídeo no padrão HEVC.

A codificação de vídeo é baseada em blocos. Com isso, é necessário dividir cada quadro em blocos quadráticos menores. Esses blocos são chamados de *Coding Tree Units* (CTUs) e podem assumir tamanhos equivalentes a 16x16, 32x32 ou 64x64 pixels. Considerando o espaço de cores YCbCr adotado, cada pixel apresenta quatro amostras de luminância e duas de croma. Tamanhos maiores de CTUs implicam em melhores taxas de compressão (SULLIVAN, *et al.*, 2012). Cada CTU é composta por uma ou mais unidades de *Coding Units* (CUs). Desta forma, a partir da CTU, é possível que cada CU seja dividida recursivamente em quatro blocos. E assim, uma árvore quadrática composta por blocos de CUs é formada, onde o tamanho de cada CU pode variar de 8x8 até o tamanho total da CTU (SULLIVAN, *et al.*, 2012). Além das CTUs e das CUs, o HEVC também tem suporte as *Processing Units* (PUs), unidades básicas de predição. E assim, cada CU pode conter uma ou mais PUs dependendo do modo de partição (SULLIVAN, *et al.*, 2012). No total, existem oito

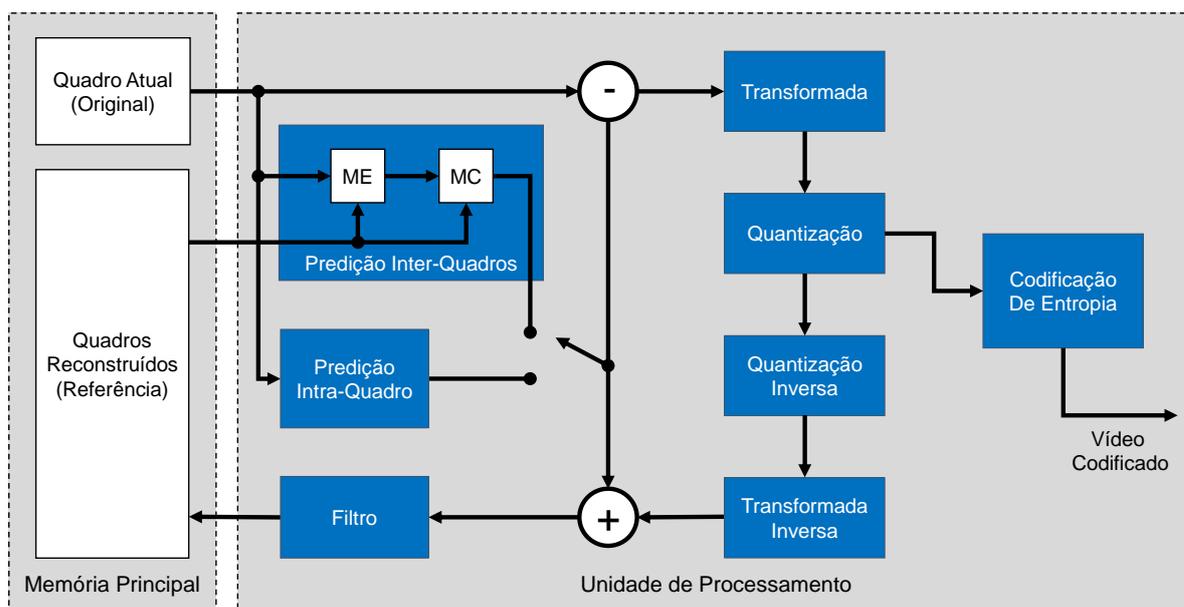


Figura 1. Diagrama de blocos simplificado de um codificador de vídeo

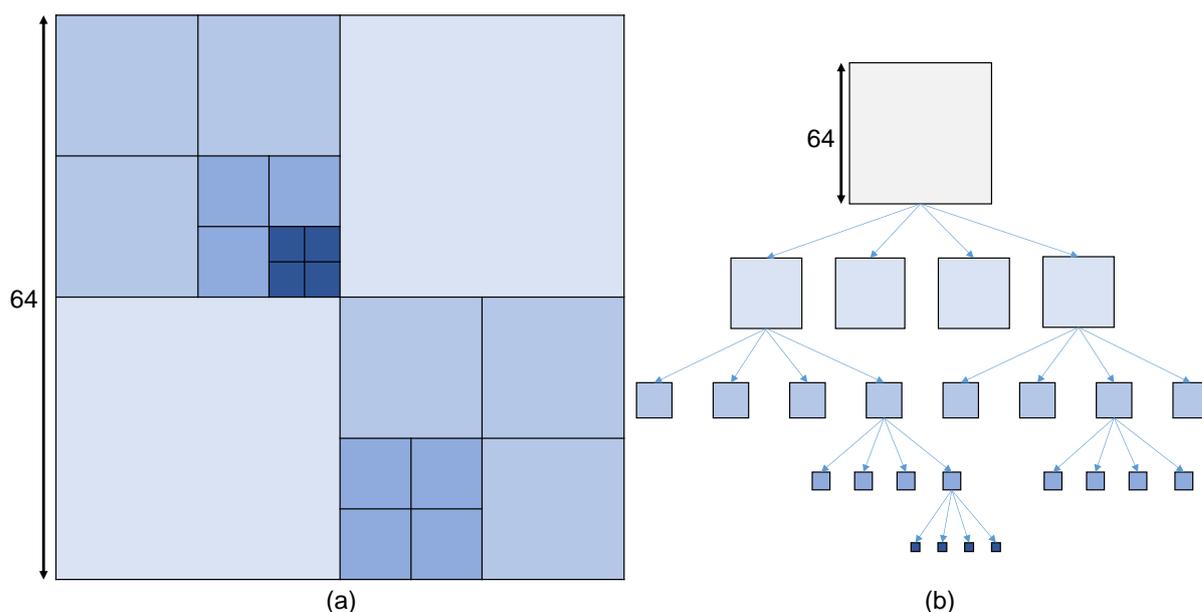


Figura 2. Particionamento de uma CTU: (a) exemplo de particionamento de uma CTU 64x64 e (b) descrição deste particionamento em forma de árvore quadrática

diferentes partições de PUs. A Figura 2 demonstra a divisão de uma CTU em CUs menores. Na Figura 2 está representada a estrutura da árvore quadrática final gerada a partir desta CTU.

O primeiro passo da codificação, conforme a Figura 1, é a etapa das predições. Esta etapa pode consistir em uma predição intra-quadro ou em uma predição inter-quadro. A predição intra-quadro é responsável por eliminar as redundâncias espaciais, presentes dentro de um mesmo quadro. A predição inter-quadro, por sua vez, é responsável por descartar as redundâncias presentes entre dois ou mais quadros. Esta predição contém o módulo da estimação de movimento, principal responsável pelos ganhos em compressão nos codificadores de vídeo.

Após a etapa das predições, os resíduos (diferenças entre o quadro predito e o quadro original) passam para os processos de transformada e quantização. O módulo da transformada, no entanto, é responsável por converter as informações de resíduos de predição do domínio espacial para o domínio das frequências. A transformada principal do HEVC é a *Discrete Cosine Transform* (DCT) (SZE, BUDAGAVI e SULLIVAN, 2014). Já o módulo da quantização, aplicado logo após a transformada, é responsável atenuar, ou até mesmo eliminar, dados de alta frequência a partir dos coeficientes obtidos na etapa da transformada. Para a realização destes cortes, há um parâmetro de quantização (*Quantization Parameter* - QP) que indica a intensidade das perdas inseridas pela quantização. Estas perdas são irreversíveis e, assim,

valores altos de QP oferecem um maior corte no número de dados utilizados para representar o vídeo, mas causam uma perda maior de qualidade (RICHARDSON, 2002).

O próximo módulo, a codificação de entropia, recebe o resíduo quantizado e aplica algoritmos de codificação sem perdas a fim de manter a qualidade do vídeo e aumentar os ganhos na taxa de compressão. As técnicas de codificação utilizadas na codificação de entropia têm como objetivo representar as informações com maior densidade de probabilidade com um menor número de bits do que informações com menor densidade de probabilidade. No HEVC, o algoritmo *Context-Adaptive Binary Arithmetic Coding* (CABAC) é utilizado nesta etapa (SZE, BUDAGAVI e SULLIVAN, 2014).

Após a etapa de quantização, são aplicadas operações de quantização inversa e transformada inversa que são adicionados aos quadros preditos para gerar os quadros reconstruídos de acordo com o processo de predição utilizado. Após, são aplicados filtros responsáveis pela remoção de artefatos inseridos pela codificação realizada em blocos. Após esta fase de reconstrução do quadro, o mesmo pode ser armazenado na memória para ser futuramente utilizado como um quadro de referência para um próximo quadro a ser codificado (SZE, BUDAGAVI e SULLIVAN, 2014).

2.1 Estimação de movimento e acessos à memória

O processo de estimação de movimento tem como objetivo a exploração da redundância temporal, presente entre dois ou mais quadros. Esta exploração ocorre para cada bloco a nível de PU no quadro atual sendo codificado (SZE, BUDAGAVI e SULLIVAN, 2014). Assim, para cada bloco no quadro atual, a ME é responsável por encontrar o bloco com maior similaridade dentro de uma área de busca de um ou mais quadros de referência. A área de busca no quadro de referência é geralmente formada ao redor do bloco colocalizado ao bloco que está sendo codificado. Este processo de encontrar um bloco com a melhor correspondência é vantajoso dentro do cenário da codificação de vídeo, pois permite que uma mesma informação presente em quadros vizinhos não precise ser armazenada mais de uma vez. Quadros vizinhos tendem a ser muito similares entre si, visto que para ter uma sensação de movimento real uma média de 24 a 30 quadros por segundo é necessária (RICHARDSON, 2002). Isto

explica o porquê de a ME ser a maior contribuinte para as altas taxas de compressão atingidas pelos codificadores de vídeo.

No entanto, o encontro da melhor correspondência entre os blocos se dá por meio de um algoritmo de estimação de movimento. Diversos algoritmos de estimação de movimento podem ser encontrados na literatura. A comparação entre estes algoritmos se dá através de métricas de qualidade como o PSNR e taxa de bits, ou *bitrate*, que é a taxa de bits necessária para que a transmissão do vídeo possa ocorrer. Além disso, os algoritmos podem ser divididos em dois grupos: os algoritmos ótimos e os algoritmos rápidos. O algoritmo ótimo *Full Search* (FS) retorna o melhor casamento e a menor taxa de bits possível para uma determinada área de busca. No entanto, este algoritmo desempenha uma busca exaustiva e lenta, comparando todos os blocos de uma dada área de busca. Algoritmos rápidos, como o *Test Zone Search* (TZS) (XIU-LI, SHENG-KUI e CAN-HUI, 2010), por outro lado, se baseiam em heurísticas para encontrar o melhor bloco correspondente, reduzindo o número de blocos comparados e tentando atingir resultados de qualidade e taxa de bits próximos aos do FS. O algoritmo TZS foi utilizado neste trabalho e será melhor discutido neste capítulo em uma subseção própria.

Para decidir qual bloco apresenta a melhor correspondência, um critério de similaridade é utilizado. Diversos critérios de similaridade são propostos na literatura, o mais utilizado, porém, é o *Sum of Absolute Differences* (SAD) (KUHN, 1999), devido a sua simplicidade e facilidade de implementação em hardware. O SAD é calculado com base na diferença absoluta entre cada pixel no bloco original e o pixel correspondente no bloco que está sendo usado para comparação. Estas diferenças são somadas para criar uma simples métrica de similaridade de bloco. Para dois blocos (B_1 e B_2) com o mesmo tamanho, em que w é a largura e h a altura, o SAD é definido por:

$$SAD = \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} |B_2(x, y) - B_1(x, y)| \quad (3)$$

Após encontrar o bloco com maior similaridade, a estimação de movimento gera um vetor de movimento que indica o deslocamento do melhor bloco encontrado em relação ao bloco colocalizado, conforme ocorre na Figura 3, que apresenta os

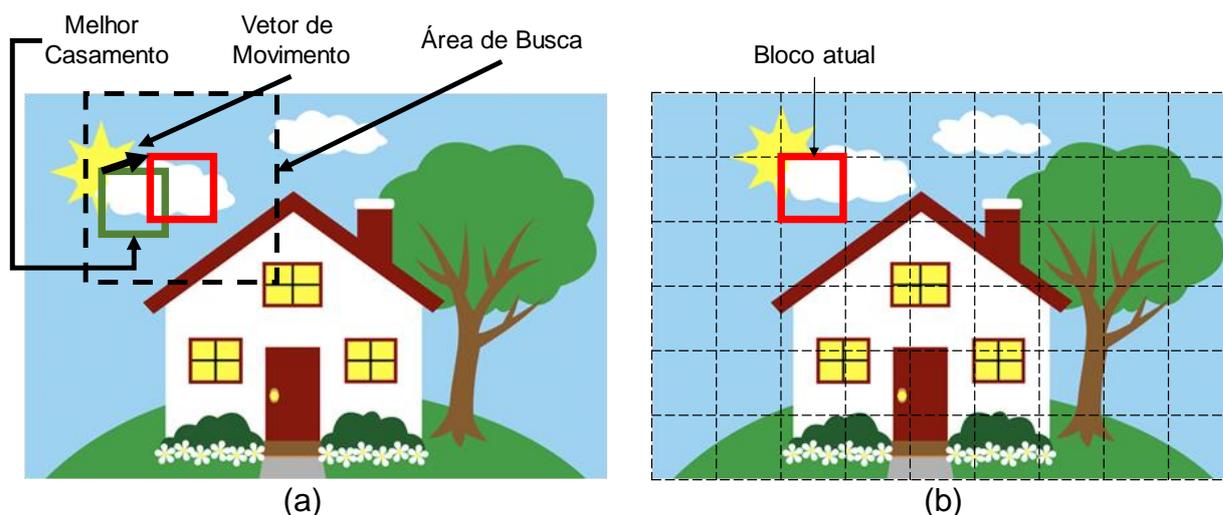


Figura 3. Representação da estimativa de movimento: (a) quadro de referência e (b) quadro atual (AMARAL, 2015)

conceitos envolvidos na estimativa de movimento. O vetor de movimento e as informações do resíduo entre o bloco sendo codificado e o bloco encontrado devem ser repassados aos demais módulos do codificador. É importante ressaltar aqui que, na etapa da ME, apenas as informações de luminância são consideradas. Os vetores de movimento definidos para as informações de luminância são posteriormente reutilizados para as informações de crominância. Esta estratégia aproveita o fato de que as informações de crominância são menos relevantes para a composição do vídeo do que as amostras de luminância (PORTO, 2012).

Ainda assim, todo o processo de estimativa de movimento demanda um grande número de acessos à memória, dado que para cada bloco sendo codificado em determinado momento, vários blocos precisam ser acessados em um ou mais quadros de referência na busca pelo bloco mais similar. Estes blocos muitas vezes compartilham as mesmas informações, e assim, um mesmo dado pode ser lido inúmeras vezes da memória externa neste processo. Os quadros de referência são armazenados no *Decoded Picture Buffer* (DPB). O DPB geralmente fica armazenado em uma memória externa, devido à sua grande quantidade de informação e, tipicamente, esta memória externa é uma *Dynamic Random-Access Memory* (DRAM) (SILVEIRA, 2015).

Como o foco deste trabalho é a redução da comunicação com a memória, o funcionamento do algoritmo de estimativa de movimento adotado precisa ser compreendido. Desta forma, a próxima seção aborda e discute o algoritmo de busca rápida *Test Zone Search* (TZS), utilizado neste trabalho.

2.2 O algoritmo Test Zone Search

Para desempenhar a etapa de estimação de movimento, o algoritmo TZS se divide em quatro etapas: Pré-Busca, Busca Inicial, Busca *Raster* e Refinamento. A sequência de execução do algoritmo é apresentada no fluxograma da Figura 4.

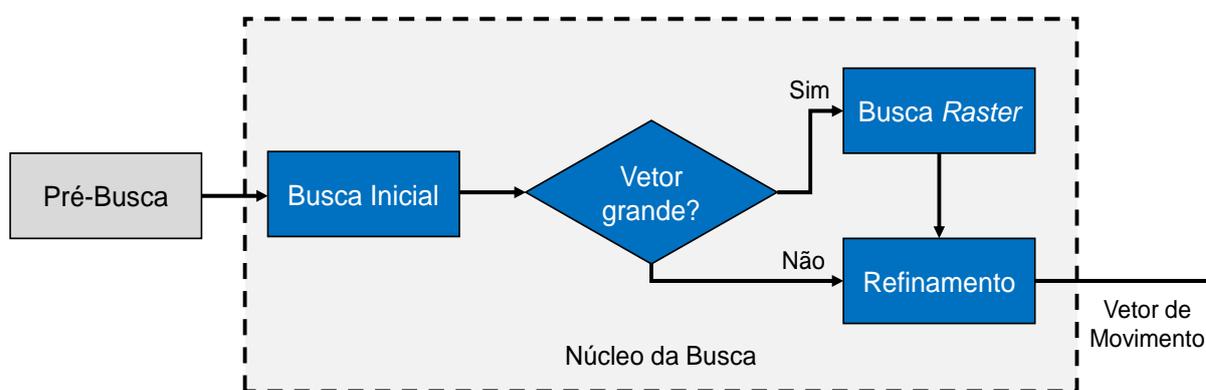


Figura 4. Fluxograma do algoritmo de busca TZS

O núcleo da busca do TZS compreende as três últimas etapas. No intuito de encontrar a melhor correspondência possível entre os blocos, o algoritmo compreende heurísticas e buscas subamostradas, evitando os mínimos locais e se aproximando dos resultados alcançados pelo algoritmo FS. No entanto, de acordo com algumas configurações internas e limites previamente definidos, a busca *Raster* pode ou não ser executada (XIU-LI, SHENG-KUI e CAN-HUI, 2010)..

A etapa de Pré-Busca se baseia em cinco preditores para encontrar a região, dentro do quadro de referência, com a melhor chance de conter o melhor casamento. Quando o encontro desta região ocorre, uma janela de busca é formada ao redor desta área e a Busca Inicial tem início. A Busca Inicial começa no centro da área de busca e se propaga em direção às suas bordas. Uma Busca *Raster* pode ser posteriormente realizada caso o melhor bloco encontrado pela Busca Inicial tenha um vetor de movimento considerado grande. Caso contrário, o algoritmo executa diretamente a fase de Refinamento, onde uma espécie de Busca Inicial com algumas alterações é realizada. As etapas do TZS são discutidas de forma mais aprofundada nas próximas subseções. O TZS utilizado neste trabalho ignora a fase de Pré-Busca, tomando sempre o bloco colocalizado como o centro da área de busca na etapa da Busca Inicial.

2.2.1 Pré-Busca

A etapa de Pré-Busca é responsável por direcionar a busca para a região, no quadro de referência, onde há a maior chance de encontro do melhor casamento. Assim, a Pré-Busca agiliza o processo realizado pelas outras etapas do TZS, pois indica a região mais promissora do quadro de referência, onde o bloco mais similar provavelmente se encontra. Esta predição é feita baseada no resultado da estimação de movimento de blocos previamente codificados. Assim, são testados cinco preditores, que são herdados de vetores de movimento de blocos previamente codificados. Estes preditores são: o bloco colocalizado (PC), o bloco imediatamente da esquerda (PE), o bloco imediatamente superior (PS), o bloco imediatamente superior direito (PSD) e a mediana (PM) dos quatro preditores anteriores (XIU-LI, SHENG-KUI e CAN-HUI, 2010), conforme demonstrado na Figura 5. Nesta figura, os blocos em cinza escuro indicam blocos já codificados e as setas em vermelho representam os vetores de movimento dos preditores utilizados nesta etapa. O melhor preditor encontrado é dado como o bloco central da Busca Inicial. A etapa de Pré-Busca favorece bons resultados de codificação, uma vez que permite que o TZS faça menos comparações e convirja para o resultado mais promissor de forma mais rápida.

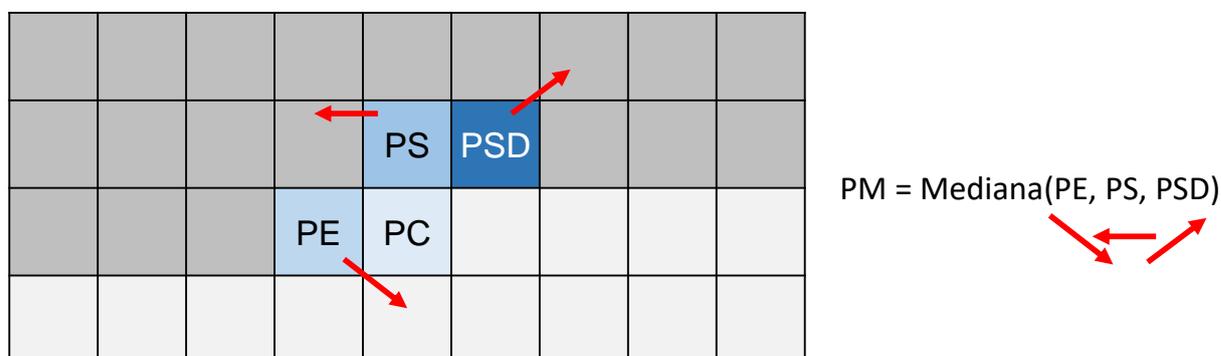


Figura 5. Ilustração da etapa de Pré-Busca e seus preditores

2.2.2 Busca Inicial

A segunda etapa é a Busca Inicial, na qual o TZS desempenha uma busca seguindo um padrão geométrico losangular ou quadrático, adotando como ponto central a melhor posição encontrada na fase da Pré-Busca (XIU-LI, SHENG-KUI e CAN-HUI, 2010). A Busca Inicial pode chegar até o tamanho máximo da área de busca, seguindo níveis de expansão com o comprimento da passada da busca crescendo em potência de dois. Uma condição impõe que a busca progrida com uma

tolerância de n níveis a frente do melhor bloco encontrado, onde o melhor bloco encontrado é inicialmente o bloco central. Como n por *default* é 3, uma tolerância de três níveis é necessária. Dito isto, existem dois critérios de parada nesta etapa de Busca Inicial que podem ser atingidos. Um deles ocorre quando a busca expande até chegar na borda da área de busca. O outro ocorre quando nenhum bloco com maior similaridade ao bloco sendo codificado é encontrado após a execução dos três níveis de expansão. A execução de três níveis de expansão é uma estratégia do TZS para evitar os mínimos locais. Desta forma, em uma área de busca com *search range* 64, o TZS faz sua expansão mínima de acordo com a Figura 6.

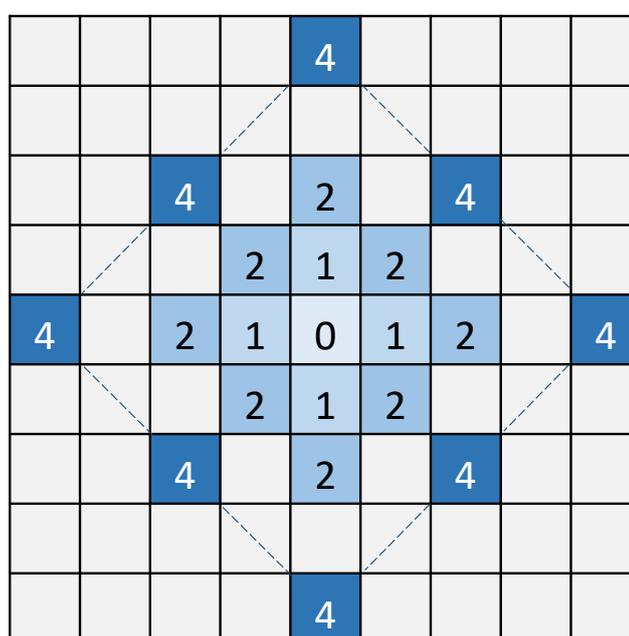


Figura 6. Busca Inicial do TZS em formato losangular

Nesta figura, o bloco zero já foi previamente comparado na Pré-Busca e considerado o melhor preditor. Assim, o TZS expande seus três níveis com passadas em potência de dois, comparando os blocos com distâncias de uma amostra do bloco central (primeira expansão), duas amostras do bloco central (segunda expansão) e quatro amostras do bloco central (terceira expansão). Seguindo a passada em potência de dois, as próximas expansões teriam distâncias equivalentes a 8, 16, 32 e 64 amostras em relação ao bloco central (XIU-LI, SHENG-KUI e CAN-HUI, 2010). A quantidade de blocos comparados, ou amostragem, em cada expansão varia de acordo com as regras abaixo:

- Se Distância (D) = 1, amostragem de quatro pontos
- Se $1 < D \leq 8$, amostragem de oito pontos

- Se $D \geq 16$, amostragem de dezesseis pontos

Ainda na Figura 6, caso nenhum dos pontos comparados apresente um resultado melhor do que o bloco central, a procura cessa e a Busca Inicial retorna o bloco 0 como resultado. Porém, caso um dos blocos da segunda expansão, por exemplo, obtenha a maior similaridade com o bloco sendo processado, a Busca Inicial irá abranger ainda mais dois níveis de expansão, contemplando os blocos distantes a 8 e 16 amostras do bloco central e respeitando a condição de tolerância de três níveis, onde a terceira expansão já foi previamente realizada. Desta forma, com o melhor bloco encontrado na segunda expansão, a Busca Inicial terá expandido para a terceira, a quarta e a quinta expansão, com blocos distantes a 4, 8 e 16 amostras do bloco central.

Após o encontro do melhor bloco na fase da Busca Inicial, o vetor de movimento deste bloco é analisado. Se este vetor for considerado grande, a Busca *Raster* é executada. Caso contrário, este bloco encontrado é tomado como o bloco central na fase de Refinamento e a Busca *Raster* é ignorada.

2.2.3 Busca *Raster*

A ocorrência da Busca *Raster* está condicionada ao fato de o vetor de movimento de saída da Busca Inicial ser grande. Para determinar se o vetor de movimento do melhor bloco encontrado na Busca Inicial é grande ou não, o tamanho deste vetor de movimento é comparado com uma constante $iRaster$ (XIU-LI, SHENG-KUI e CAN-HUI, 2010). Esta constante tem valor igual a 5, como padrão, no software de referência do HEVC, o HEVC *Test Model* (HM) versão 16.6 (HM, 2016). Assim, se o melhor bloco encontrado na Busca Inicial estiver a uma distância equivalente a 1, 2 ou 4 amostras do bloco central ou for o bloco central, ou seja, se estiver na área mínima da Busca Inicial, a Busca *Raster* não é executada e o algoritmo TZS passa direto para a execução da fase de Refinamento. Caso contrário, o algoritmo entende que o melhor casamento ainda não foi encontrado e realiza a Busca *Raster* dentro da área de busca, em uma segunda tentativa de se aproximar de uma região com maior probabilidade de encontrar o melhor casamento. A Busca *Raster* se caracteriza como uma varredura subamostrada, comparando blocos na área de busca com o bloco sendo codificado, respeitando uma distância horizontal e vertical entre blocos igual ao valor da $iRaster$. Por ser uma busca subamostrada, o encontro do melhor casamento não é garantido e uma fase de Refinamento ainda é necessária. A Busca *Raster*,

quando realizada, se torna a de maior custo computacional, pois ela obrigatoriamente compara e busca um maior número de amostras na área de busca do que as outras etapas do TZS. Na Figura 7, assume-se uma área de busca com 7x7 blocos e uma *iRaster* com valor igual a 2. Deste modo, somente os blocos em cor azul serão comparados ao bloco que está sendo codificado. O bloco de saída da Busca *Raster* é colocado como o bloco central da fase de Refinamento.

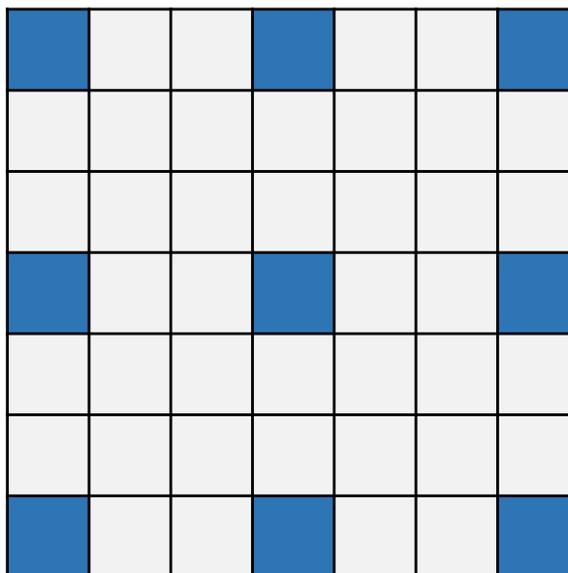


Figura 7. Ilustração da Busca *Raster*, com *iRaster* = 2

2.2.4 Refinamento

A etapa de Refinamento é a última a ser realizada pelo algoritmo TZS. Esta etapa assume o melhor bloco encontrado até então (proveniente da Busca Inicial ou da Busca *Raster*, quando executada), como bloco central de sua busca e desempenha o mesmo padrão de busca da etapa de Busca Inicial. No entanto, na fase de Refinamento, o centro da área de busca é atualizado com o melhor resultado obtido na última iteração e esta fase possui um menor nível de tolerância, tipicamente dois. A fase de Refinamento, assim como a Busca Inicial, possui duas condições de parada. Uma delas é quando a expansão atinge a borda da área de busca. A outra, quando o TZS não encontra um melhor casamento após a execução dos dois níveis de expansões. Na Figura 8, uma ilustração da etapa de Refinamento é apresentada. A expansão mínima da fase de Refinamento corresponde a duas expansões a frente do melhor bloco encontrado, proveniente da etapa executada anteriormente (Busca Inicial ou *Raster*). Na Figura 8(a), após a expansão de dois níveis a frente do bloco 0,

um melhor bloco candidato foi encontrado na primeira expansão (bloco 1 na cor preta da figura à esquerda). Assim, na próxima iteração, este bloco é escolhido como o bloco central e o Refinamento itera novamente, conforme a Figura 8(b).

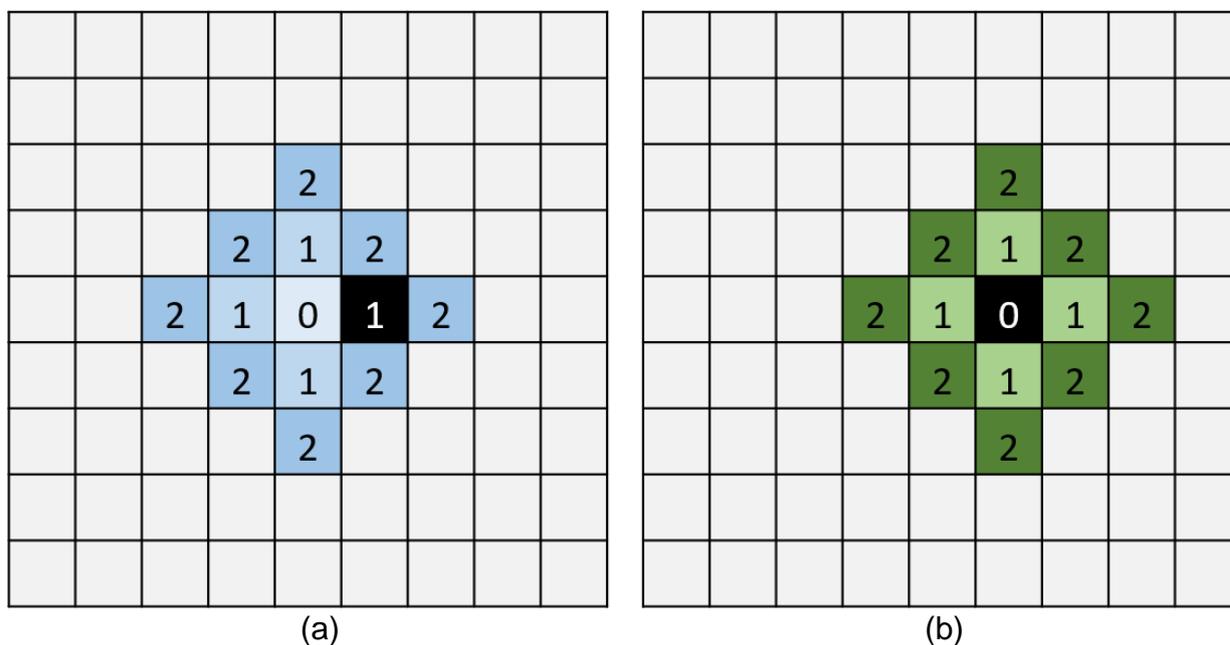


Figura 8. Ilustração da fase de Refinamento: (a) primeira iteração e (b) segunda iteração

2.3 Considerações finais do capítulo

Este capítulo reuniu uma série de conceitos necessários para a melhor compreensão do processo de codificação de vídeo em si. Assim, foram apresentadas as definições de vídeo digital, pixel e o espaço de cores YCbCr, onde cada pixel traz informações de luminância, cromaância azul e cromaância vermelha. Como as informações de cromaância são menos significativas ao sistema visual humano, estas podem ser representadas com um número menor de bits do que as informações de luminância. Esta redução faz com que os sistemas de codificação de vídeo se beneficiem com uma quantidade menor de dados que precisam ser armazenados, sem causar grandes perdas na qualidade visual.

Posteriormente, o capítulo abordou os três tipos de redundância que são explorados pela codificação de vídeo, sendo eles a redundância espacial, a redundância temporal e a redundância entrópica. Enquanto as duas primeiras redundâncias são descartadas na etapa das predições, a redundância entrópica é explorada na etapa da codificação de entropia. Devido à alta similaridade entre os quadros vizinhos, impulsionada pela taxa de quadros por segundo na transmissão do vídeo, a redundância temporal é a mais significativa na codificação do vídeo e isso faz

com que a estimação de movimento se torne a maior responsável pelas altas taxas de compressão atingidas pelos padrões de codificação de vídeo.

A etapa da estimação de movimento foi apresentada de forma mais aprofundada em uma subseção própria, onde o processo de busca pelo bloco de maior similaridade e geração do vetor de movimento foram detalhados. Como explicado nesta subseção, a ME é responsável por um grande volume de acessos à memória externa. Isto ocorre porque o processo de encontro do melhor casamento entre o bloco que está sendo codificado e blocos presentes em quadros de referência exige que um grande número de dados presentes na memória externa seja comparado ao bloco sendo codificado. O número de comparações e acesso a dados na memória externa impacta diretamente no consumo de energia de sistemas de codificação de vídeo.

No entanto, o número de comparações está diretamente relacionado com o algoritmo de busca de estimação de movimento adotado. Com isso, na subseção 2.2 o algoritmo TZS, utilizado e estudado neste trabalho, teve suas etapas discutidas de forma detalhada. Esse algoritmo se divide em quatro etapas: Pré-busca, Busca Inicial, Busca *Raster* e Refinamento e aplica heurísticas e buscas subamostradas na tentativa encontrar o melhor casamento com um menor número de comparações.

Por fim, a comunicação com a memória no cenário da codificação de vídeo gera um alto consumo de energia. O capítulo 3 deste trabalho discute a ligação entre número de acessos e consumo de energia na codificação de vídeo, apresenta os trabalhos que propõe estratégias para redução de acessos à memória e, por fim, faz uma análise da distribuição dos acessos do TZS dentro de uma área de busca e dentre regiões de um mesmo quadro.

3 CONSUMO DE ENERGIA NA ESTIMAÇÃO DE MOVIMENTO E ANÁLISE DOS ACESSOS À MEMÓRIA

A estimação de movimento, ou ME, corresponde ao módulo mais complexo de um codificador de vídeo (GRELLERT, BAMPI e ZATT, 2016). Sendo o módulo mais complexo, a ME também é a maior consumidora de tempo dentro do processo de codificação de vídeo, chegando a atingir 80% do tempo total de computação em um codificador HEVC (BOSSSEN, et al., 2012). Além disso, durante o processamento da ME, a energia consumida relacionada aos acessos à memória contribui com aproximadamente 90% do total de energia consumida pela ME (ZATT, et al., 2011).

Estes acessos ocorrem durante a busca, no DPB (*Decoded Picture Buffer*), pelos blocos de maior similaridade durante a codificação. O processamento de um vídeo *High Definition* (HD) 1080p (1920x1080 pixels), considerando apenas um quadro de referência, com blocos de tamanho 64x64 amostras e área de busca composta por 256x256 amostras, ocasionaria a busca na memória externa de 506 áreas de busca para codificar apenas um quadro. O uso de um maior número de quadros de referência gera melhores resultados de codificação, mas acrescenta um custo significativo no número de acessos à memória e na complexidade do codificador (BOSSSEN, et al., 2012).

Para estimar o consumo de energia envolvido nas operações de leitura e de escrita da memória externa é preciso considerar a quantidade de dados envolvida no acesso à memória e o custo de energia relacionado a cada acesso. Uma forma de quantificar o consumo de energia envolvido neste processo é utilizar algumas fórmulas, como as apresentadas em (AMARAL, 2015) e que são discutidas neste trabalho. As Equações 4 e 5 quantificam o consumo de energia (CE) relacionado aos acessos de leitura (Re) e escrita (Wr) para uma quantidade D de palavras de tamanho de 32 bits quando o método tradicional, ou seja, sem estratégias de redução de acessos, é utilizado.

$$CE_{DRAM/T_{Re}} = E(DRAM_{Re}) * D(Algorithm_k) \quad (4)$$

$$CE_{DRAM/T_{Wr}} = E(DRAM_{Wr}) * D(Frame) \quad (5)$$

Ainda nas Equações 4 e 5, E é o custo de energia das operações de escrita e de leitura em um tipo específico de memória, $Algoritmo_k$ é o algoritmo k escolhido para desempenhar a busca na etapa de estimação de movimento. No escopo deste trabalho, k corresponde ao algoritmo TZS. Por fim, $Frame$ representa o tamanho total de um quadro. Assim, considerando vídeos HD 1080p, $Frame$ seria equivalente a 1920×1080 pixels. O consumo de energia total para o método tradicional é calculado conforme a Equação 6.

$$CE_T = CE_{DRAM/T_{Re}} + CE_{DRAM/T_{Wr}} \quad (6)$$

O alto consumo de energia no método tradicional implica na necessidade do uso de estratégias para redução dos acessos a memória. Estratégias possíveis de serem utilizadas para este fim são o reuso de dados, a compressão de quadros de referência e o ajuste dinâmico da área de busca. O reuso de dados armazena uma área de um quadro de referência em uma memória rápida, tipicamente uma *Static Random Access Memory* (SRAM), reduzindo, assim, o número de acessos de leitura aos dados na memória externa e, conseqüentemente, a energia consumida neste processo. Alguns métodos de reuso de dados são encontrados na literatura, como os tipos *Level A*, *Level B*, *Level C* ou *Level D* (TUAN, CHANG e JEN, 2002), ou ainda do tipo *Level C+* (CHEN, *et al.*, 2006). Considerando os requisitos de hardware e o comportamento do sistema de codificação de vídeo, o *Level C* é a estratégia mais praticável (AMARAL, 2015).

Compressores de quadros de referência atuam tanto na escrita quanto na leitura da memória. Para isto, os quadros de referência passam por um processo de compressão, que pode ou não inserir perdas no processo, antes de serem armazenados na memória externa e são descomprimidos quando o dado é solicitado pela estimação de movimento. Compressores com perdas, como em (CHENG, TSENG e CHEN, 2009), costumam atingir melhores resultados de compressão do que os compressores que não degradam a qualidade do vídeo, como os apresentados em (SILVEIRA, 2015) e (POVALA, 2015). No entanto, um processo de compressão de quadros de referência com perdas não é desejável, dado que os erros gerados em um quadro de referência são propagados para todo o processo de codificação, prejudicando a qualidade final do vídeo. As técnicas de reuso de dados e compressão

de quadros de referência podem ser usadas em conjunto, como em (AMARAL, 2015), potencializando os ganhos destas duas técnicas.

Todavia, enquanto estas duas técnicas aumentam relativamente o custo de hardware, uma terceira estratégia se propõe a reduzir a largura de banda de memória por meio de algoritmos de ajuste dinâmico de área de busca (*Dynamic Search Range* - DSR). Este ajuste dinâmico é geralmente realizado com base nos tamanhos dos vetores de movimento encontrados para a codificação do quadro anterior e nos resultados de SAD.

3.1 Trabalhos relacionados

Dado o foco deste trabalho no projeto e gerenciamento de uma memória de busca para a estimação de movimento, somente trabalhos abordando o reuso de dados e algoritmos DSR serão discutidos na próxima seção. No entanto, é importante notar que as técnicas de compressão de quadros de referência podem ser usadas em conjunto com a proposta deste trabalho de dissertação.

3.1.1 Reuso de dados

As técnicas de reuso de dados empregam memórias SRAM para implementar hierarquias de memória por meio de *scratchpad memories* (SPMs). Menos custosas em termos de tempo de acesso e energia (JASUJA, 2018), as SRAMs permitem o acesso ao dado mais rapidamente e diminuem o gargalo de energia e de desempenho quando comparadas às DRAMs. Diversos trabalhos propõem e discutem as estratégias de reuso de dados. No entanto, as soluções mais relevantes são as apresentadas nos trabalhos (TUAN, CHANG e JEN, 2002) e (CHEN, *et al.*, 2006) muito utilizadas como referência por outros trabalhos na literatura.

O trabalho (TUAN, CHANG e JEN, 2002) apresenta quatro *Levels* de reuso de dados e considera dois níveis de redundância de informações que devem ser reutilizadas: regiões sobrepostas entre blocos candidatos vizinhos dentro de uma mesma área de busca (*Levels* A e B) e regiões sobrepostas entre áreas de busca de blocos candidatos imediatamente vizinhos (*Levels* C e D). No entanto, enquanto o *Level/C* armazena a área de busca em uma SPM, o *Level/D* necessita armazenar todo o quadro, tornando esta solução muito mais custosa em hardware do que o *Level/C*.

O *Level/C+* é proposto em (CHEN, *et al.*, 2006) Esta estratégia é uma expansão do *Level/C* e considera o reuso de regiões sobrepostas das áreas de buscas entre

blocos imediatamente vizinhos em duas direções: horizontal e vertical. Embora apresente melhores resultados de eficiência quando comparado às estratégias propostas em (TUAN, CHANG e JEN, 2002), o *Level C+* exige uma ordem de codificação que não corresponde ao que ocorre nos padrões de codificação de vídeo, o que implica na necessidade de um aumento de hardware para que esta técnica possa ser utilizada. A medida de eficiência utilizada pelo autor considera a largura de banda da memória principal e o tamanho da memória SPM.

A solução proposta em (JAKUBOWSKI e PASTUSZAK, 2010) consiste na implementação de uma hierarquia de memória de dois níveis, que combina os *Levels C* e *D*. A estratégia multinível adotada reduz a largura de banda da memória externa em cerca de 77% nas operações de leitura para vídeos com resolução HD 720p (1280x720 pixels), além de reduzir o tamanho da memória interna em 93% com o uso do *Level C*.

O trabalho apresentado em (GRELLERT, *et al.*, 2011) consiste em uma estratégia multinível baseado no *Level C+*. Este trabalho envolve dois níveis de hierarquia de memória: uma cache local e um *buffer* local. Esta estratégia de reuso de dados com dois níveis consegue atingir uma redução de largura de banda de até 90% para acessos de leitura, quando utilizada na codificação de vídeos HD 1080p.

Em (SAMPAIO, *et al.*, 2014) uma combinação de SPMs privadas e sobrepostas é feita, viabilizando o reuso de dados dentro e entre diferentes núcleos que executam *threads* HEVC de forma concorrente. O trabalho também desenvolveu um método estatístico para dimensionar e projetar a organização das SPMs, junto com uma política adaptativa de gerenciamento dinâmico de leituras de memória, a fim de gerenciar os estados de energia dessas memórias por meio de *power gating*. Os experimentos foram realizados com cinco tamanhos diferentes de área de busca, quatro vídeos 1080p e dois vídeos com resolução 2K (2500x1600 pixels). Os resultados mostraram que houve uma redução energética de até 61% no consumo de energia total quando comparado com o uso de SPMs *Level C*.

3.1.2 Algoritmos para DSR

A técnica proposta em (SAPONARA e FANUCCI, 2004) explora a correlação espacial e temporal dos vetores de movimento e consiste em, a partir de todos os vetores de movimento obtidos para o quadro anterior, calcular o deslocamento máximo encontrado. Esse deslocamento máximo indicará o tamanho da área de

busca para o quadro atual, onde uma busca exaustiva é realizada. A técnica também apresenta mecanismos para evitar distorções elevadas, ampliando a área de busca, caso necessário. Os resultados apresentados para o algoritmo FS apontam que este trabalho reduz o consumo de potência em cerca de 70% a 90% para vídeos com resoluções QCIF (176x144 pixels) e CIF (352x288 pixels) a um custo muito baixo de perda de qualidade de vídeo. Resultados menos expressivos são apresentados para outros dois algoritmos rápidos em (SAPONARA, *et al.*, 2006).

Uma estratégia conhecida como *Adaptive Search Range Motion Estimation* (ASRME) é apresentado em (SONG, *et al.*, 2007), onde ele estima o tamanho do vetor de movimento médio do quadro anterior para adaptar o tamanho da área de busca do quadro atual. Para propor esta técnica, os autores consideraram as alterações na qualidade de vídeo, medida em PSNR e em *bitrate* quando o tamanho da área de busca sofria variações. Assim, o ASRME é composto de duas partes: a primeira consiste em calcular o vetor de movimento médio de cinco quadros anteriores. A segunda parte do ASRME verifica se o melhor SAD encontrado é menor do que um SAD limite previamente definido. Se a resposta para este teste for positiva, a busca encerra. Caso não seja, a área de busca é levemente ampliada e o teste é novamente realizado. A área de busca pode ser ampliada em até duas vezes. Os resultados mostram que o algoritmo atingiu resultados variados na redução da complexidade computacional, alternando entre 13,41% a 73,41% de redução, para 12 vídeos de resolução desconhecida, e atingindo, na média, 55% de redução da complexidade computacional, com baixa perda de qualidade.

Diferente da abordagem proposta por (SAPONARA e FANUCCI, 2004), a técnica proposta em (CHANG e CHUNG, 2010) apresenta uma estratégia que considera o vetor de movimento máximo encontrado no quadro anterior, mas também o SAD médio. Então, a área de busca inicial de um quadro começa com o tamanho do vetor máximo do quadro anterior. Considerando apenas a correlação temporal, a ocorrência de grande movimentação de um objeto entre dois quadros vizinhos causará erros de predição grandes. Por isso, o algoritmo proposto também verifica o espaço de correlação espacial, onde o vetor de movimento máximo dos blocos vizinhos é considerado. Mesmo assim, a fim de evitar mínimos locais, o algoritmo define o SAD médio por bloco no quadro anterior como um limite. Quando o SAD dos blocos vizinhos é maior do que este limite, ou seja, quando apresenta uma distorção maior, a área de busca volta a ter seu tamanho original, o maior possível. Resultados

deste trabalho mostram que, considerando o algoritmo FS, uma redução de acessos de 80% a 90% é atingida para vídeos com resolução CIF e HD 1080p, com baixa perda em PSNR.

O trabalho de (JUNG, KIM e KYUNG, 2010) é o primeiro a apresentar um algoritmo de DSR que considera uma área de busca não quadrada. Além disso, o trabalho associa este algoritmo com uma técnica de compressão de quadros de referência com perdas (Kim, 2009) e com a estratégia de reuso de dados *Level C* (Tuan, 2002). Para a definição do DSR proposto, é apresentada uma relação entre a taxa de compressão, o tamanho do vetor de movimento, o tamanho da área de busca e o tráfego de dados. A partir disto, uma área de busca de formato diamante é formada e armazenada em uma memória com estratégia *Level C*. Os resultados deste trabalho mostram que ele atinge uma média de 80% de redução de largura de banda em relação ao algoritmo FS para dez vídeos HD 1080p, com perda de 0,02 dB no PSNR. O trabalho não apresenta resultados de consumo de energia.

Em (DAI, *et al.*, 2012), há uma verificação de que a distribuição de Laplace pode não descrever bem a distribuição dos tamanhos dos vetores de movimento. O estudo realizado também aponta que cerca de 80% dos vetores de movimento apontam para o bloco colocalizado. Desta forma, o artigo propõe o uso da distribuição de Cauchy, para melhorar a efetividade na predição da distribuição dos tamanhos dos vetores de movimento. Esta distribuição é utilizada para descrever os tamanhos dos vetores de movimento do quadro anterior. No quadro sendo codificado, um refinamento é aplicado, a fim de garantir a eficácia do tamanho da área de busca. Uma redução de custo computacional de 95% é atingida, com baixa variação no PSNR.

O trabalho de (JIA, *et al.*, 2013) calcula que a probabilidade do vetor de movimento ótimo ser encontrado é maior em áreas de buscas em formato diamante do que em áreas de buscas quadráticas. Conforme os demais trabalhos, a área de busca é então ajustada conforme os tamanhos de vetores de movimento encontrados no quadro anterior. Os resultados mostram que esta técnica atinge quase 80% de redução de complexidade computacional na média, com perda de 0,03 dB no *Bjontegaard Delta Rate* (BD-Rate). Para os testes, foram utilizados vídeos com resolução 416x240, 832x480 e HD 720p e o algoritmo FS foi utilizado como parâmetro de comparação.

Em (DU, *et al.*, 2014), há uma proposta de um algoritmo *Adaptive Search Range* (ASR) que, diferente dos outros trabalhos, considera os encontros de melhor casamento das PUs, e não apenas os valores do tamanho dos vetores de melhor movimento. Assim, para PUs com baixa movimentação, o algoritmo calcula um tamanho de área de busca menor e para PUs com alta movimentação, áreas de buscas maiores são adotadas. Para chegar no tamanho de área de busca indicado, PUs vizinhas temporais e espaciais são consideradas. Este algoritmo salva 42% do tempo de codificação da ME com degradação de 0,023 dB no *Bjontegaard Delta PSNR* (BD-PSNR) e aumento de 0,7% no BD-RD quanto embarcado no algoritmo TZS.

Existem duas propostas em (CHIEN, LIAO e YANG, 2014) para reduzir o tempo de codificação no padrão HEVC. Uma delas é um melhoramento do mecanismo de seleção do melhor vetor de movimento e a outra, um algoritmo de área de busca adaptativa. O algoritmo *Adaptive Motion Search Range* (AMSR) se propõe a utilizar as diferenças do vetor de movimento como um critério primário para justificar se o tamanho da área de busca de uma determinada PU pode ser reduzida. Para isto, os autores utilizam o vetor de movimento do bloco colocalizado do quadro anterior para prever o tamanho do atual vetor de movimento e, assim, determinar um tamanho de área de busca. Com esta técnica, os autores reduziram a complexidade em 43%, com perda de eficiência de 0,019 dB no BD-PSNR.

O relacionamento entre o preditor de vetor de movimento, a diferença entre PUs e o tamanho da área de busca é levado em consideração em (LI, *et al.*, 2015). Neste processo, a codificação se divide em duas partes: a etapa de aprendizado e a etapa de predição. No processo de aprendizado, o algoritmo coleta alguns dados relacionados ao tamanho dos vetores de movimento que são futuramente utilizados para formar uma árvore no processo de predição. E assim, por meio do algoritmo *k-nearest neighbor* (k-NN), a etapa de predição calcula o melhor tamanho da área de busca. Esta técnica reduz 91% da complexidade da ME e traz um aumento de 0,17% no BD-Rate.

Por fim, em (JI, *et al.*, 2015), os autores modelaram uma métrica de desvio do preditor de vetor de movimento para prever a relação entre o tamanho da área de busca e a diferença do vetor de movimento, de acordo com as diferenças temporais e espaciais de blocos vizinhos. O trabalho também apresenta um algoritmo de controle de ASR para gerenciar a complexidade computacional enquanto maximiza a

qualidade da codificação, colocando mais esforço computacional em blocos com tamanho de vetor de movimento maior e menos esforço em blocos com vetor de movimento menor. Resultados para este trabalho apontam que esta técnica pode atingir de 50% a 90% de redução de computação, chegando a aumentar o PSNR em até 0,3 dB quando os recursos computacionais são restritos a um certo nível.

Embora todos os trabalhos apresentados nesta subseção apliquem algoritmos de DSR, muitos ainda exploram o algoritmo FS, levando a custos desnecessários de acessos à memória. Também não há, nestes trabalhos, um estudo que relacione a quantidade e a localização dos acessos dentro da área de busca quando um algoritmo rápido é utilizado, como a análise que será apresentada na próxima seção. Esta análise permite uma maior compreensão do padrão de comportamento dos acessos e, com isto, leva a ajustes de redução e aumento mais eficientes no tamanho da área de busca. Diante deste cenário, é possível também criticar a falta de uma abordagem relacionada com desligamento de setores da memória, a fim de reduzir não apenas os dados acessados na estimação de movimento como também o consumo de energia do codificador como um todo.

Uma análise da distribuição dos acessos do TZS é feita na próxima seção. Esta análise possibilitou um melhor entendimento do comportamento do algoritmo dentro de uma área de busca e o conhecimento das regiões mais e menos acessadas.

3.2 Análise da distribuição dos acessos

Adotando o TZS como o algoritmo de busca da ME, este trabalho realizou uma análise dos acessos às amostras presentes dentro de uma área de busca. A etapa de Pré-Busca do TZS foi desabilitada, já que o fato de que a área de busca pode não ser sempre formada ao redor do bloco colocalizado dificulta implementações em hardware e impede o uso eficiente de estratégias de reuso de dados. Além disso, como este trabalho foca na estimação de movimento, este módulo foi isolado em todo o trabalho. Com isso, técnicas como o *early skip* e a ocorrência de blocos intra em quadros inter também foram desabilitadas. Ainda, a configuração *Low Delay P* foi adotada para todo o trabalho desenvolvido nesta dissertação.

Assim, para quantificar o total de amostras acessadas dentro de uma área de busca e a posição de acesso, foram consideradas seis sequencias de vídeo HD 1080p: *BasketballDrive*, *BQTerrace*, *Cactus*, *Kimono*, *ParkScene* e *Tennis* e os QPs 22, 27, 32 e 37. O tamanho de bloco de 64x64 amostras e um *search range* (SR) [-64,

+63] foram utilizados, totalizando uma área de busca de 192x192 amostras, já que a área de busca é dada por $(2 \times SR_h + BS) \times (2 \times SR_v + BS)$ amostras, onde BS é o tamanho de um lado do bloco e h e v indicam as direções horizontal e vertical. Além disto, modificações foram feitas no software de referência do HEVC, o HM 16.6, para que as informações necessárias para este estudo fossem obtidas através de um *trace*.

A estrutura do *trace* implementado no HM está presente na Figura 9. Os dados que compõem este *trace* envolvem as posições referentes às posições do bloco em processamento ($PosXBlocOrig$, $PosYBlocOrig$) no quadro original, que está sendo codificado, o deslocamento do bloco candidato em relação a este bloco em processamento ($DeslXBlocCand$, $DeslYBlocCand$) e o tamanho do bloco ($TamBloco$). Dado que blocos não-quadráticos são utilizados em uma escala muito menor do que os blocos quadráticos (SULLIVAN, *et al.*, 2012), os blocos não-quadráticos foram desabilitados na etapa da ME.

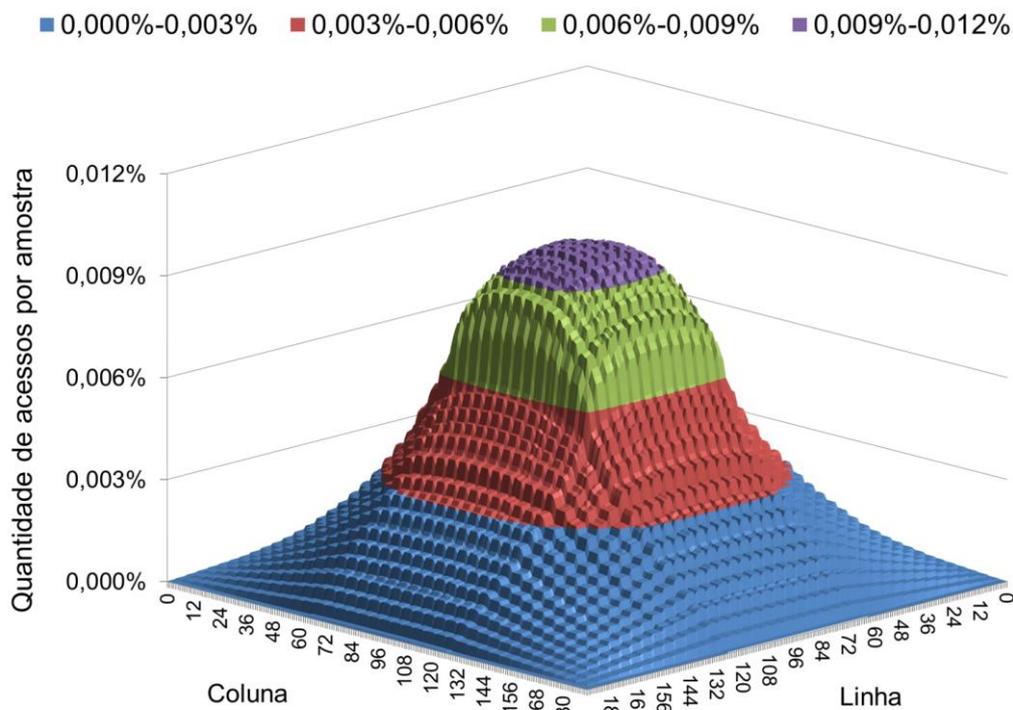
```

1.  struct trace {
2.      short PosXBlocOrig;
3.      short PosYBlocOrig;
4.      char DeslXBlocCand;
5.      char DeslYBlocCand;
6.      char TamBloco;
7.  }
```

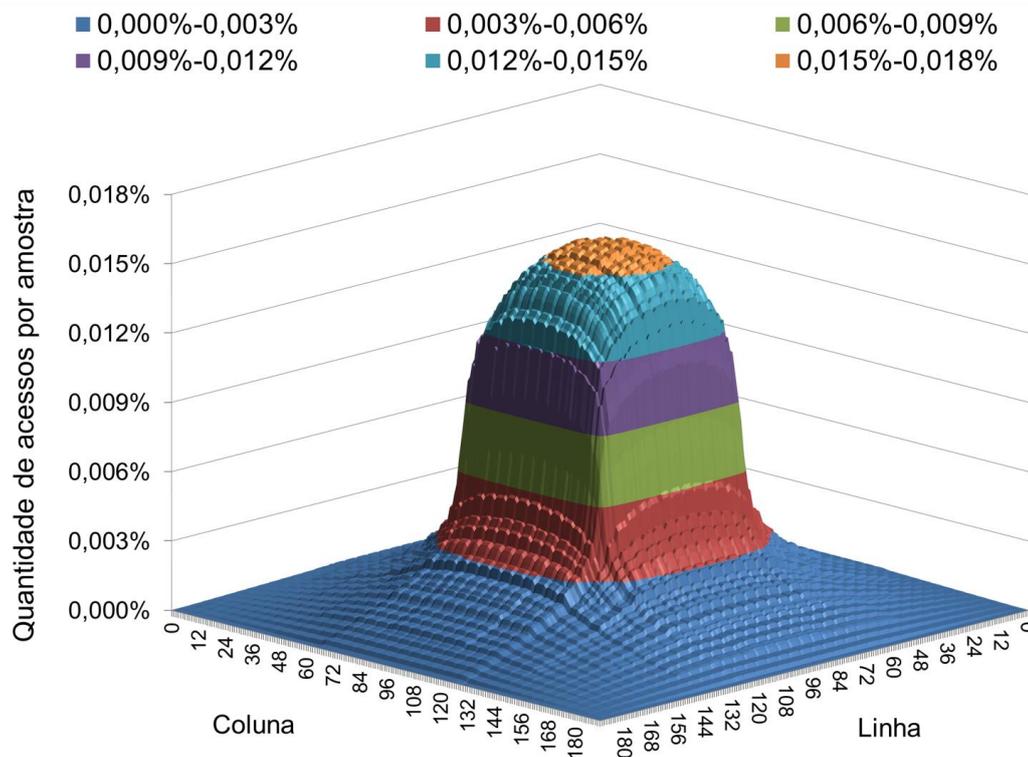
Figura 9. Estrutura do *trace* utilizado

Com as informações obtidas pelo *trace* foi contabilizado o número de acessos feitos a cada amostra na área de busca durante o processamento de 100 *frames* de cada um dos seis vídeos avaliados, a fim de verificar o comportamento dos acessos que ocorre dentro da janela de busca. A Figura 10 apresenta os seis histogramas referentes a estes acessos, gerados pela média dos QPs de cada vídeo.

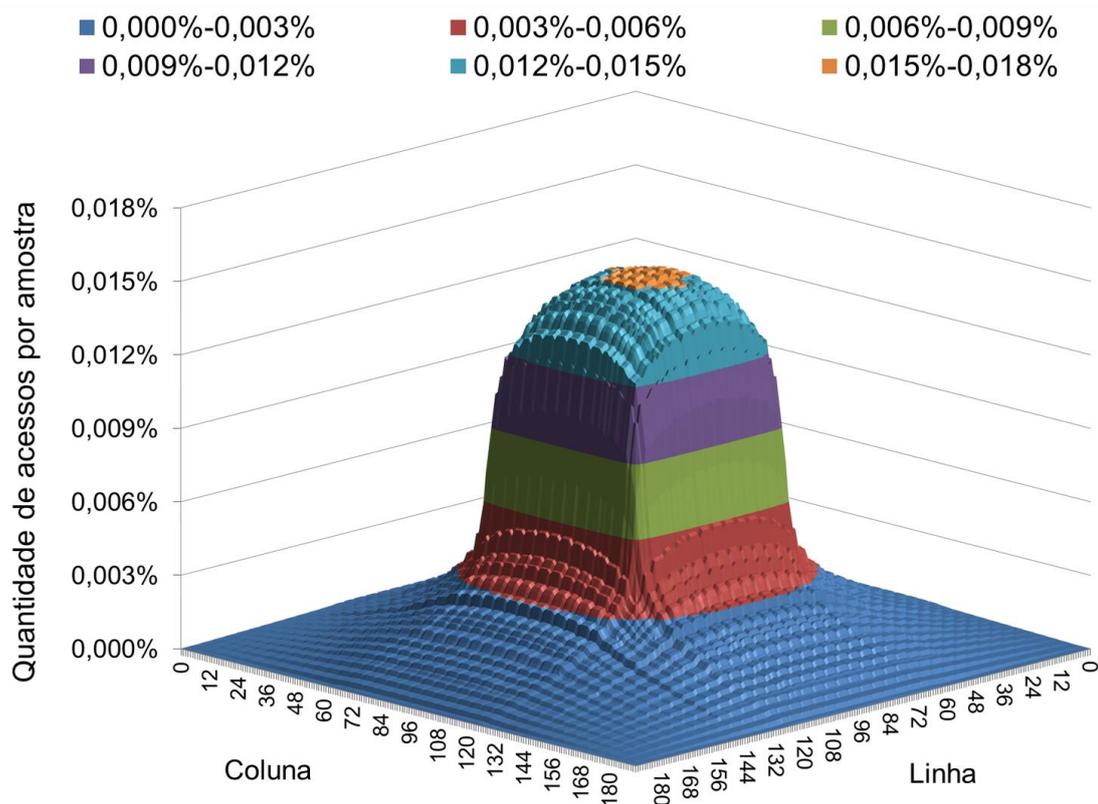
Os histogramas da Figura 10 mostram, dentro de uma área de busca de 192x192 amostras, o percentual de acesso de cada amostra. Somando o total de ocorrências de acessos às amostras, chegamos em 100%. Em uma versão colorida desta figura, as regiões em tons de azul, mais próximas da borda da área de busca, representam as amostras que receberam menos acessos, enquanto as regiões em tons de vermelho, verde e roxo, mais centrais e mais próximas do topo dos histogramas, representam as amostras que receberam o maior número de acessos.



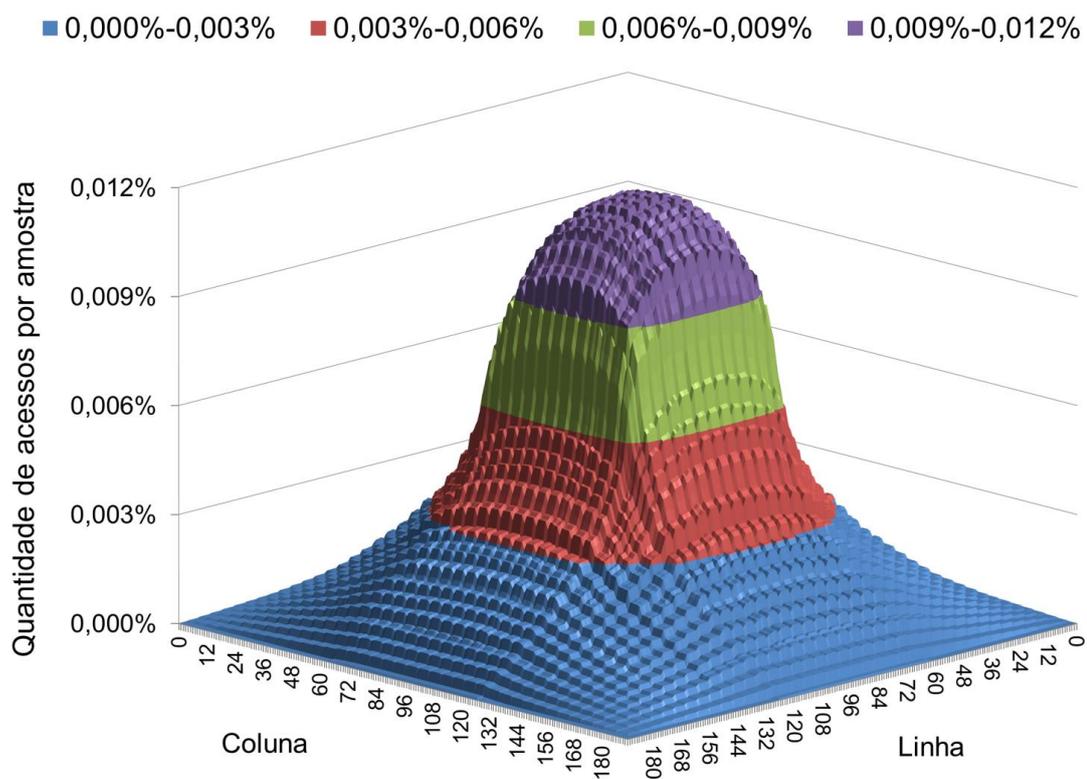
(a) BasketballDrive



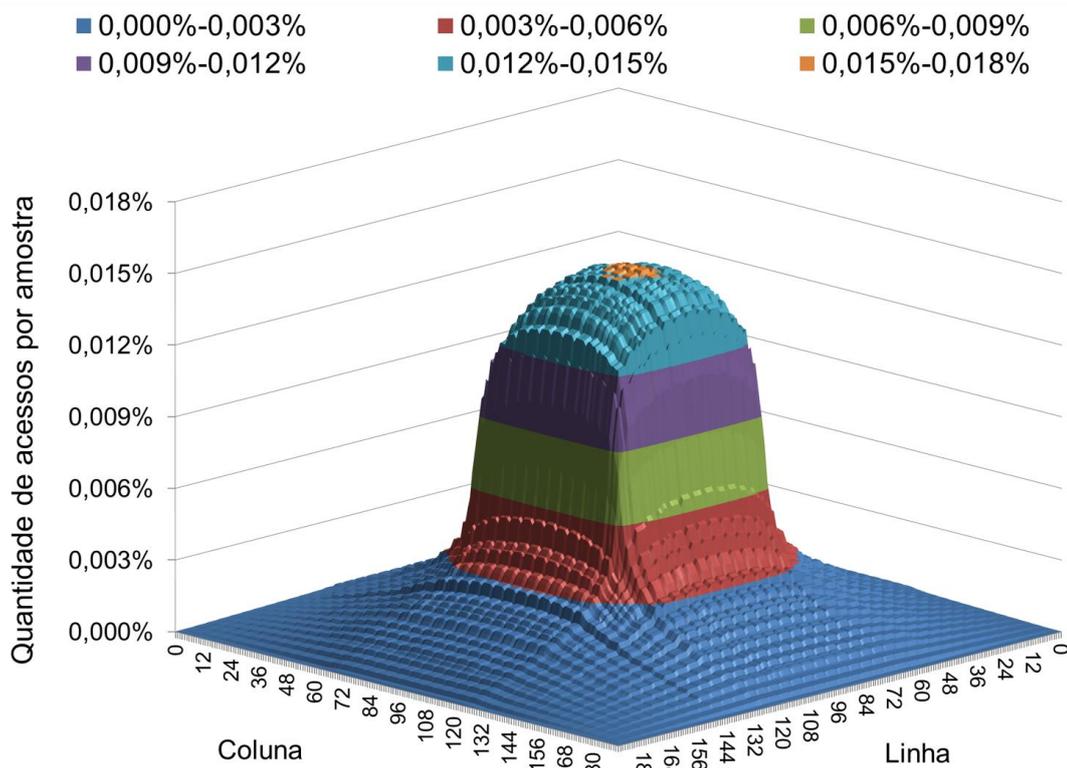
(b) BQTerrace



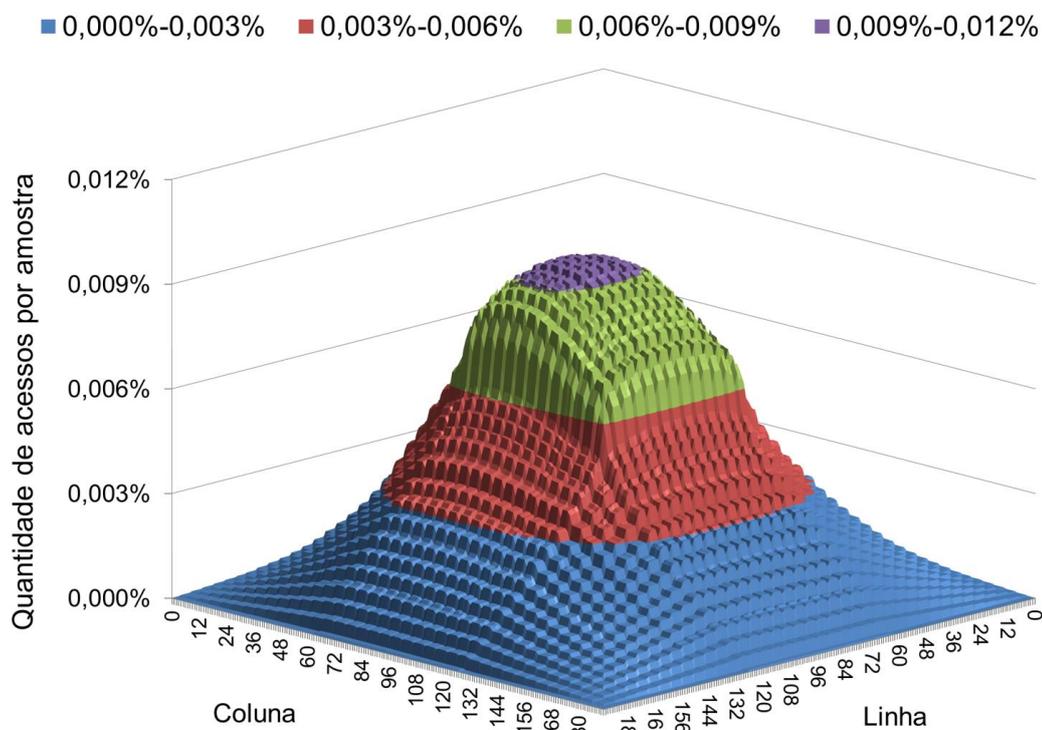
(c) Cactus



(d) Kimono



(e) ParkScene



(f) Tennis

Figura 10. Comportamento dos acessos na área de busca de cada vídeo

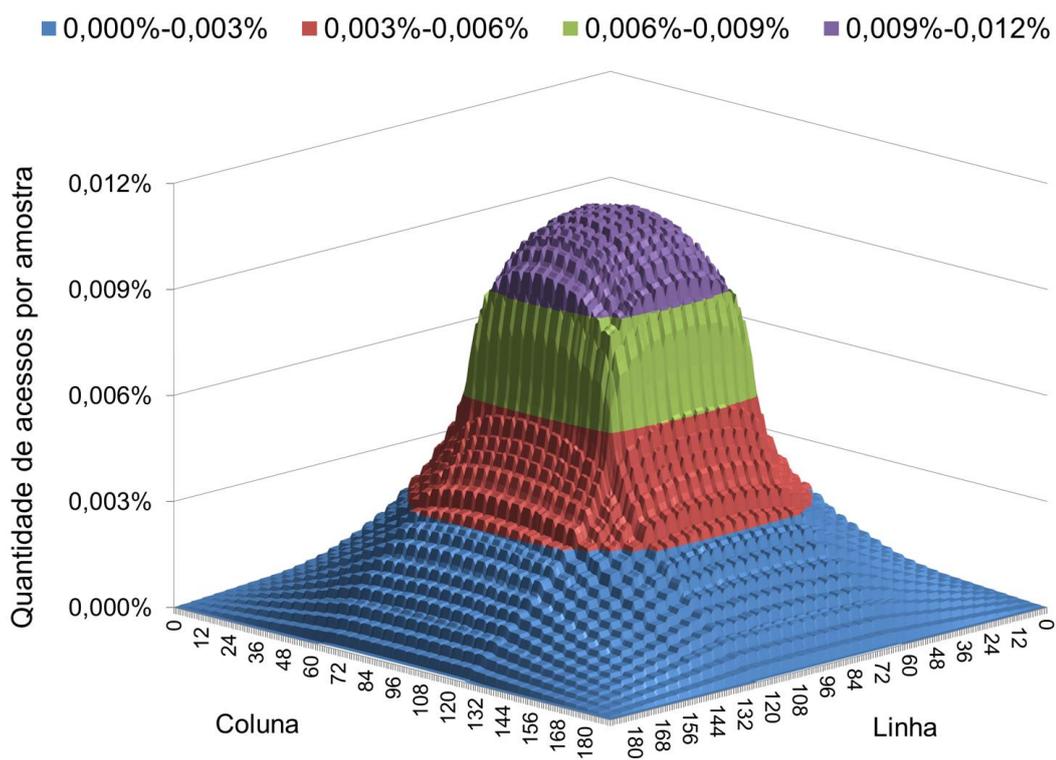
Por meio da Figura 10 nota-se que a região mais acessada na área de busca de todos os vídeos é a região central, onde se localiza o bloco colocalizado ao bloco em processamento e de onde parte a expansão do TZS durante a fase de Busca Inicial. Ademais, fica visível na figura que vídeos com alta movimentação ou textura como *BasketballDrive*, *Kimono* e *Tennis*, apresentaram um acesso maior nas regiões mais próximas a borda da área de busca do que vídeos que apresentam menor movimentação ou textura, como os vídeos *BQTerrace*, *Cactus* e *ParkScene*, onde os acessos apresentaram uma concentração central e poucos acessos ocorreram fora desta região.

A ocorrência de uma maior quantidade de acessos em regiões mais próximas às bordas da área de busca em vídeos de alta movimentação é explicada pelo fato de que, quando o vídeo apresenta uma movimentação alta, o bloco com maior similaridade tende a estar localizado em um ponto mais afastado do bloco colocalizado, ou seja, da região central. Esta distância faz com que o algoritmo expanda mais a sua busca e realize um maior número de comparações. Se a busca atingir um determinado nível de expansão, a Busca *Raster* poderá ocorrer, como já explicado anteriormente, aumentando ainda mais o número de comparações e acessos realizados na memória.

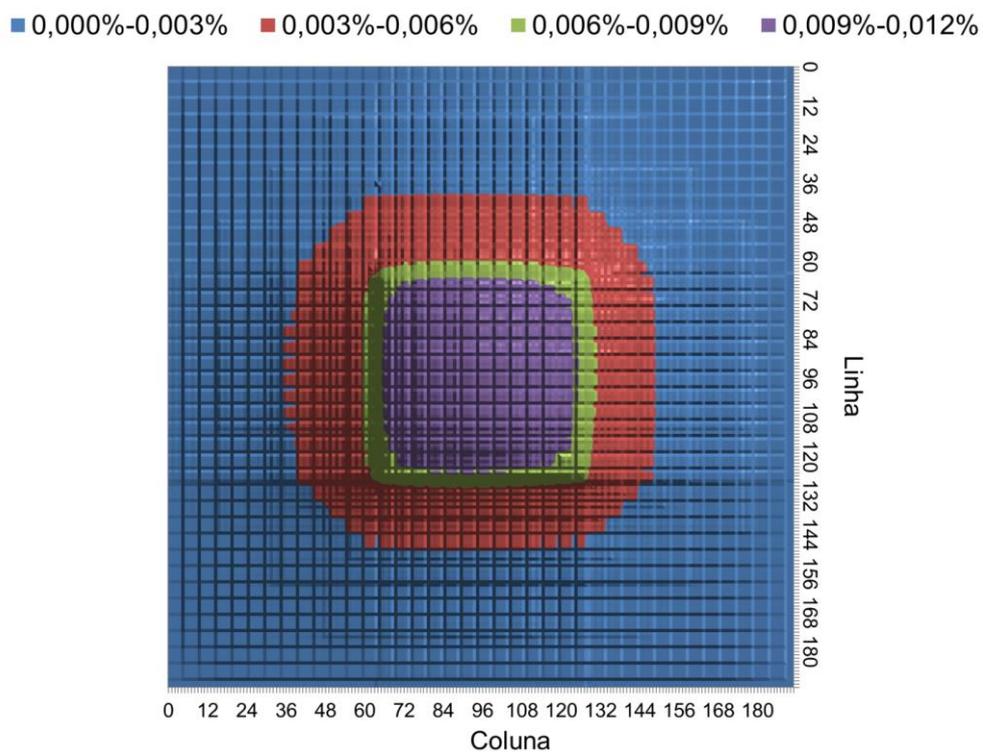
A Figura 11 ilustra o histograma referente à média dos acessos, com as visões lateral (a) e superior (b). Esses histogramas reafirmam a ocorrência da maior parte dos acessos na região central da área de busca. No entanto, na visão superior do histograma, é notável que uma região considerável da área de busca é representada por amostras que recebem poucos acessos. Assim, embora a maioria dos acessos se concentre na região central da área de busca, a grande maioria das amostras recebe poucos acessos. Essa relação entre número de acessos e região da área de busca onde esses acessos ocorrem é apresentada no gráfico da Figura 12.

Com a relação apresentada no gráfico da Figura 12, nota-se que as amostras que representam 50% dos acessos dentro da área de busca se encontram em uma região que corresponde a apenas 15,81% do tamanho total desta janela. Além disso, uma região que corresponde a um terço da área de busca compreende as amostras que receberam apenas 5% do total de acessos.

A análise destes resultados também proporcionou um maior entendimento do que ocorre em diferentes regiões de um mesmo quadro, como será abordado na próxima subseção.



(a)



(b)

Figura 11. Comportamento médio dos acessos durante a codificação dos vídeos analisados: (a) visão lateral e (b) visão superior

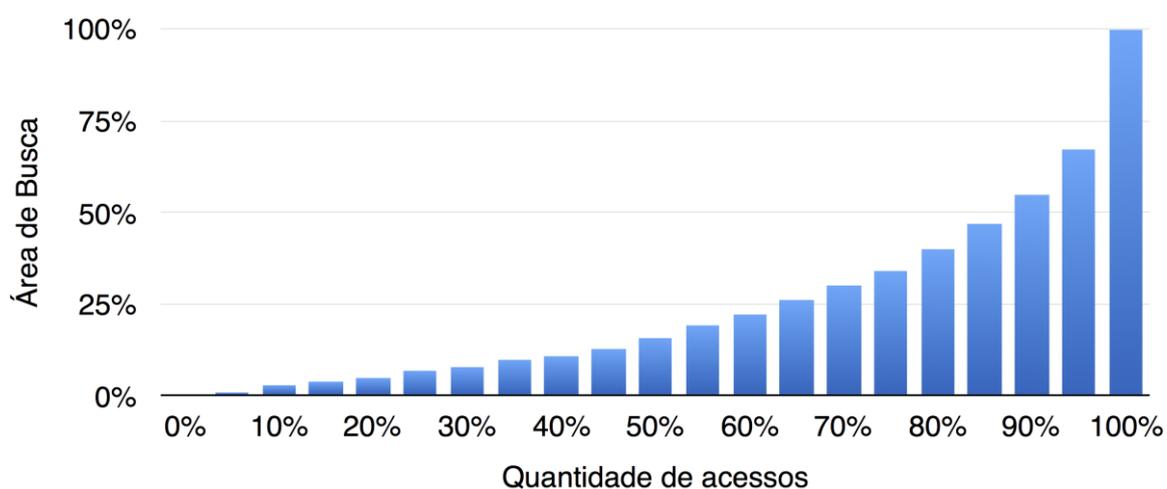


Figura 12. Relação entre os dados mais acessados e a área onde se encontram

3.3 Análise dos acessos em regiões do mesmo quadro

Essa subseção visa discutir o que ocorre em regiões diferentes de um mesmo vídeo. O trabalho (AMARAL, 2015), que serviu de motivação para este trabalho, propôs um modelo para avaliação do consumo de energia em codificadores de vídeos digitais. A validação deste modelo envolveu uma estratégia que surgiu a partir da junção de um compressor de quadros de referência e o esquema de reuso de dados *Level C*. No entanto, é preciso considerar que os dados armazenados no *Level C* e que representam toda uma área de busca para codificar uma determinada CTU nem sempre são acessados. A partir da Figura 13 até a Figura 18, são mostrados os histogramas referentes a quantidade de acessos feitos para codificar a primeira linha de CTUs de cada vídeo analisado neste trabalho.

Vídeos como *BasketballDrive*, *Kimono* e *Tennis* tiveram acessos na maioria ou em todas as suas amostras, como pode ser visto na Figura 13, na Figura 14 e na Figura 15. Isso ocorre porque esses vídeos são de alta movimentação. Em vídeos de alta movimentação há uma maior dificuldade no encontro do melhor casamento, pois o bloco mais similar tende a ser encontrado em um local mais afastado do centro, de onde parte a busca do TZS. Este caso leva a uma maior quantidade de expansões em nível durante a Busca Inicial e a ocorrência de Busca *Raster*. Por outro lado, nas imagens da Figura 16, da Figura 17 e da Figura 18, referentes aos vídeos *BQTerrace*, *Cactus* e *ParkScene*, notam-se claramente regiões de baixo ou nulo acesso às amostras nas vistas laterais e superiores. Como estes são vídeos de baixa movimentação, o melhor casamento tende a ser encontrado próximo ao centro da

área de busca. Isso faz com que a Busca Inicial realize um número pequeno de expansões e consiga convergir mais rápido para o bloco mais similar. Em alguns casos, a Busca *Raster* não precisa ser executada. Isso faz com um número significativo de amostras dentro da área de busca não sejam acessadas.

Além disso, como observado nas figuras, embora vídeos possam ser classificados como sendo de baixa ou alta movimentação, é importante notar que dentro de um mesmo vídeo cada CTU pode apresentar características diferentes na codificação. Desta forma, é um equívoco considerar que vídeos de baixa movimentação sempre vão acessar apenas a área central da área de busca e que vídeos de alta movimentação sempre vão fazer acessos até a borda da janela de busca. Assim, cada região de um vídeo apresenta características diferentes e essas características devem ser consideradas durante a codificação, a fim de se obter a melhor eficiência energética e de qualidade.

Para buscar as amostras da memória externa e armazená-las na SPM, há um custo de energia. Desta forma, se fosse possível não realizar a busca e armazenamento de amostras que não serão acessadas posteriormente pelo algoritmo da ME, os resultados alcançados no trabalho (AMARAL, 2015) de redução do consumo de energia poderiam ser ainda maiores. O trabalho aqui proposto vai ao encontro desta ideia, que reforça a validade de estudos voltados para o comportamento dos acessos feitos às amostras dentro da área de busca. Este comportamento serve como base para viabilizar ajustes na janela de busca, a fim de reduzir a quantidade de dados acessados pela ME e manter a eficiência de codificação a um nível aceitável. No entanto, ajustes que reduzem o tamanho da janela de busca implicam no fato de que alguns dados não poderão ser acessados pelo algoritmo da ME. Isto é positivo quando estes dados não trazem informações relevantes ao encontro do melhor casamento. No entanto, quando regiões de áreas de busca são removidas, é possível que dentre estas regiões esteja também o bloco que representa o melhor casamento. Dito isto, com a análise da distribuição realizada neste capítulo, foram feitos cortes fixos na janela de busca, a fim de avaliar os impactos na codificação quando as regiões que contém um certo número de dados menos acessados são removidas da área de busca original. Este estudo se encontra no capítulo 4.

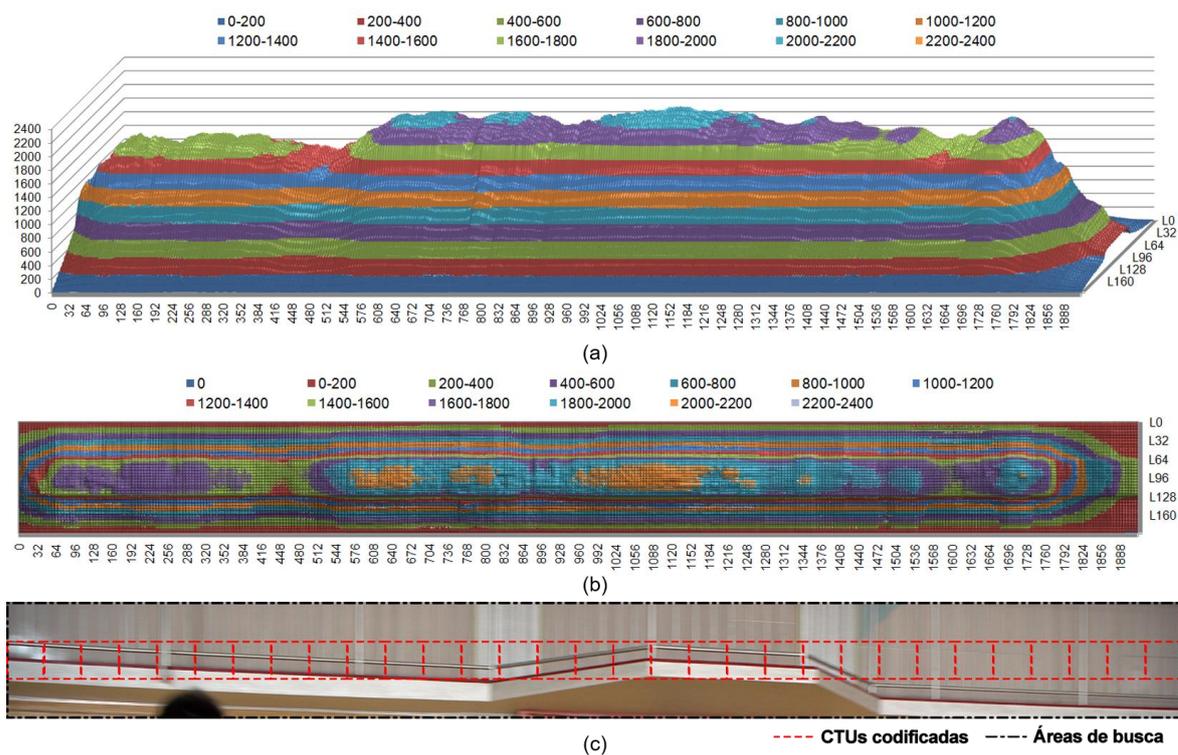


Figura 13. Análise dos acessos para codificar uma linha de CTUs do vídeo *BasketballDrive*: (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo

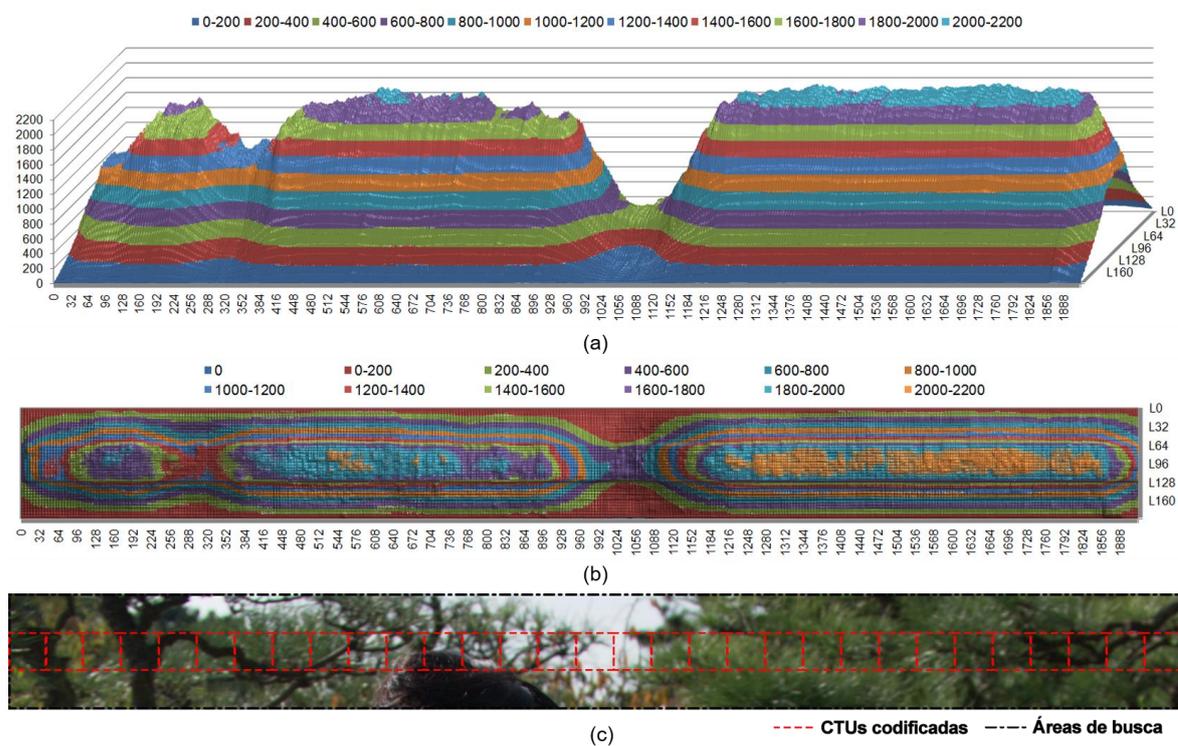


Figura 14. Análise dos acessos para codificar uma linha de CTUs do vídeo *Kimono*: (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo

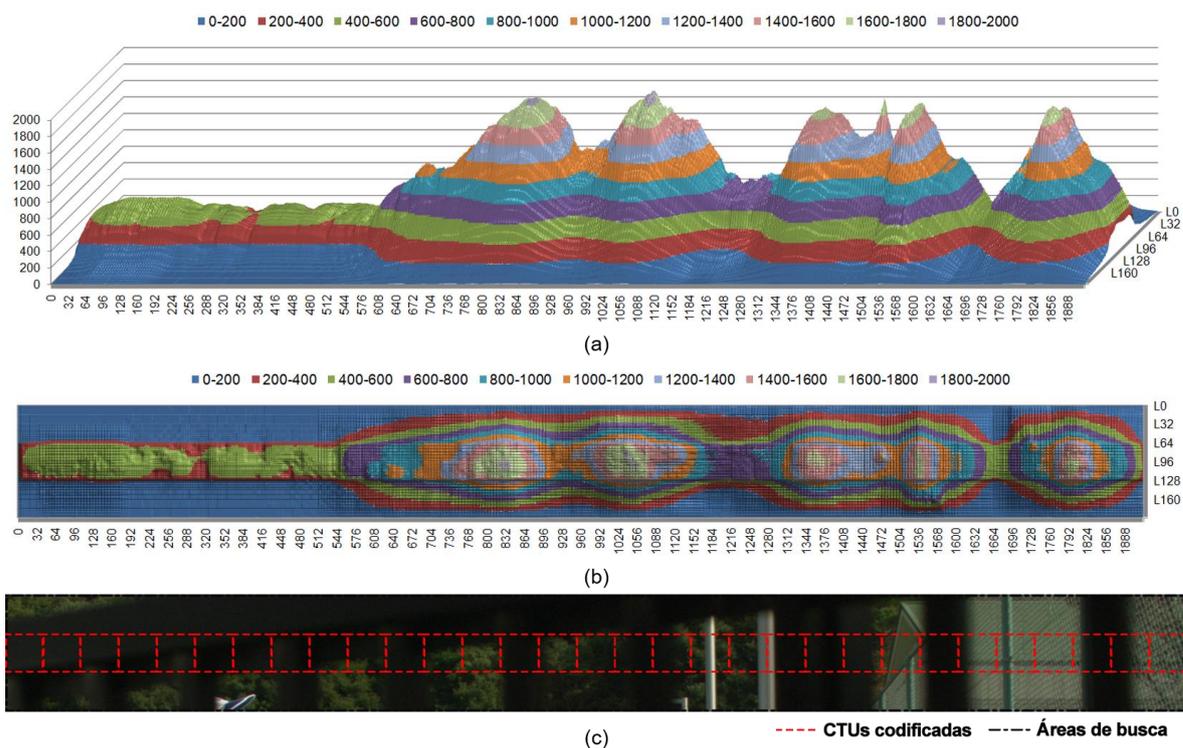


Figura 15. Análise dos acessos para codificar uma linha de CTUs do vídeo *Tennis*: (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo

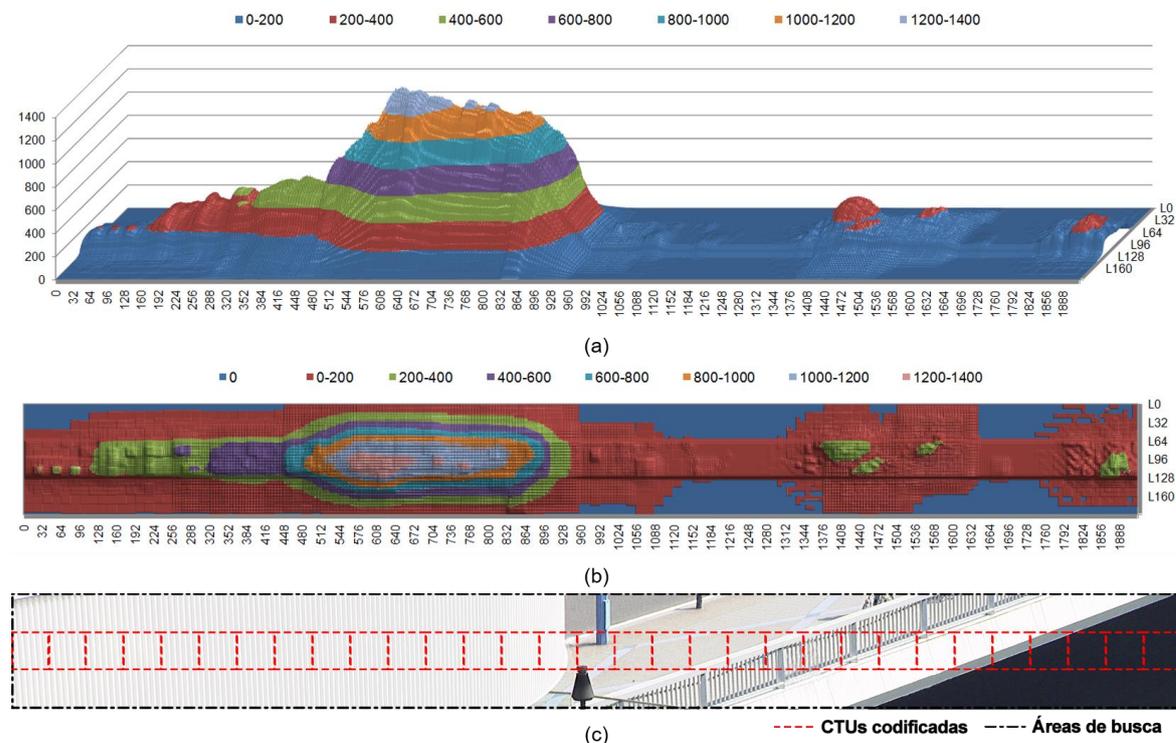


Figura 16. Análise dos acessos para codificar uma linha de CTUs do vídeo *BQTerrace*: (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo

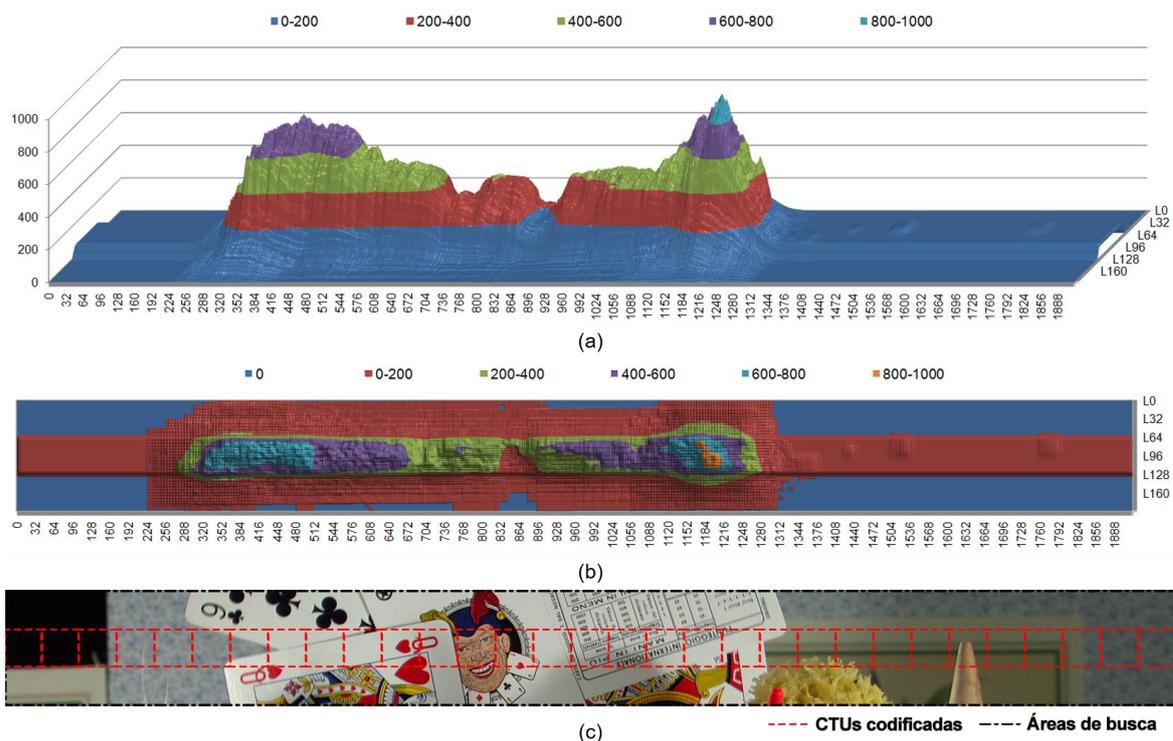


Figura 17. Análise dos acessos para codificar uma linha de CTUs do vídeo *Cactus*: (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo

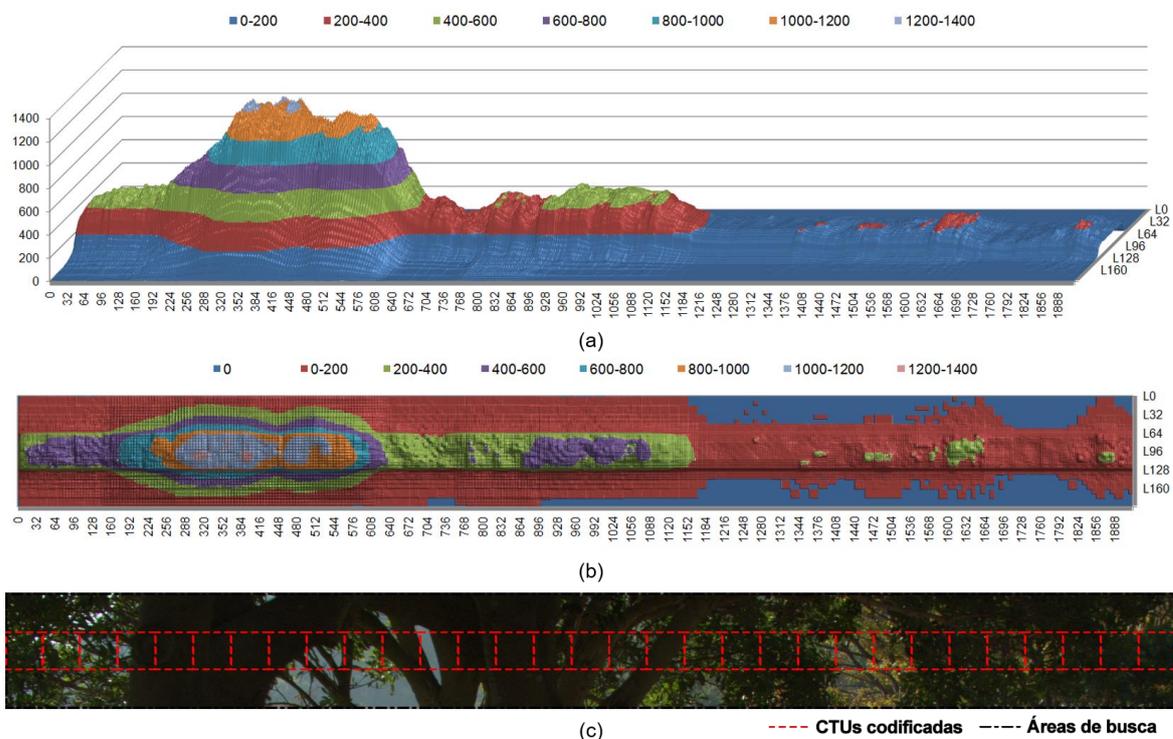


Figura 18. Análise dos acessos para codificar uma linha de CTUs do vídeo *ParkScene*: (a) vista lateral, (b) vista superior e (c) região correspondente no vídeo

3.4 Considerações finais do capítulo

Este capítulo abordou os principais problemas relacionados aos numerosos acessos realizados na memória externa na etapa de estimação de movimento, que culminam em um alto consumo de energia. Isto ocorre principalmente porque os dados buscados estão armazenados no DPB, fazendo com que a comunicação com a memória domine o consumo de energia, uma vez que o custo das operações de leitura nesta memória é geralmente superior ao custo dos acessos no núcleo de processamento do codificador de vídeo.

Com isto, foi discutido como se dá o cálculo do consumo de energia total quando nenhuma estratégia de redução de acessos é utilizada. Como o consumo neste caso é alto, foram discutidas também estratégias que se propõe a reduzir a quantidade de dados acessados na memória externa durante a codificação. Assim, as técnicas de compressão de quadros de referência, reuso de dados e algoritmos de ajuste dinâmico foram abordadas. Como o foco deste trabalho se relaciona com as últimas duas técnicas apresentadas, trabalhos que seguem estas linhas foram discutidos. No entanto, os trabalhos com algoritmos DSR relacionados se baseiam em características como valor do SAD ou tamanho do vetor de movimento para ajustar o tamanho da área de busca e não abordam um estudo aprofundado sobre a relação de quantidade de acessos e localização destes acessos dentro da área de busca durante a ME. Outro ponto negativo nestes trabalhos é que eles não abordam a redução de consumo de energia, ponto principal deste trabalho.

Neste capítulo também foi realizado, na seção 3.2, um estudo estatístico dos acessos do TZS foi realizado. A partir deste estudo foi possível observar que uma grande região da área de busca, a parte mais externa e mais próxima as bordas, recebe um número ínfimo de acessos e que grande parte dos acessos ocorrem na parte mais central da área de busca. Este comportamento foi apresentado por todos os seis vídeos abordados neste estudo. Com este estudo, foi possível compreender melhor a ocorrência dos acessos no estudo de caso do trabalho (AMARAL, 2015), que foi abordado na seção 3.3.

As observações realizadas nas seções 3.2 e 3.3 embasaram o desenvolvimento das soluções baseadas em cortes fixos na área de busca e, posteriormente, nas soluções de gerenciamento dinâmico da estimação de movimento centrado em uma memória de busca, que serão apresentadas nos próximos capítulos.

4 ESTIMAÇÃO DE MOVIMENTO POR MEIO DE CORTES FIXOS NA ÁREA DE BUSCA

Neste trabalho, as soluções desenvolvidas são aplicadas a um sistema que une também o compressor de quadros de referência *Double Differential Reference Frame Compressor* (DDRFC) (SILVEIRA, *et al.*, 2015) e o reuso de dados *Level C*. A Figura 19 demonstra esse sistema em alto nível. No esquema apresentado pela figura e proposto neste trabalho, a SPM onde atua o *Level C* permite o *power gating* de suas células, possibilitando que a área de busca original possa assumir tamanhos menores. Este sistema é usado tanto para os cortes fixos quanto para o gerenciamento dinâmico, apresentado no capítulo 5. Neste capítulo serão avaliados diferentes tamanhos fixos de área de busca, considerando o estudo realizado no capítulo anterior.

O gráfico na Figura 12, proveniente do estudo da distribuição dos acessos às amostras do TZS no capítulo 3, mostra que 95% dos dados mais acessados se encontram em uma região correspondente a 66,65% do tamanho total da área de busca. Como 95% é uma quantidade bastante significativa, o primeiro teste com corte fixo realizado foi a remoção da região restante dessa área de busca. Desta forma, a região que abrange os dados 5% menos acessados foi removida da área de busca original, que possui *search range* (SR) [-64, +63]. Assim, toda uma região que representa 33,35% do tamanho total da área de busca e que contempla estes acessos

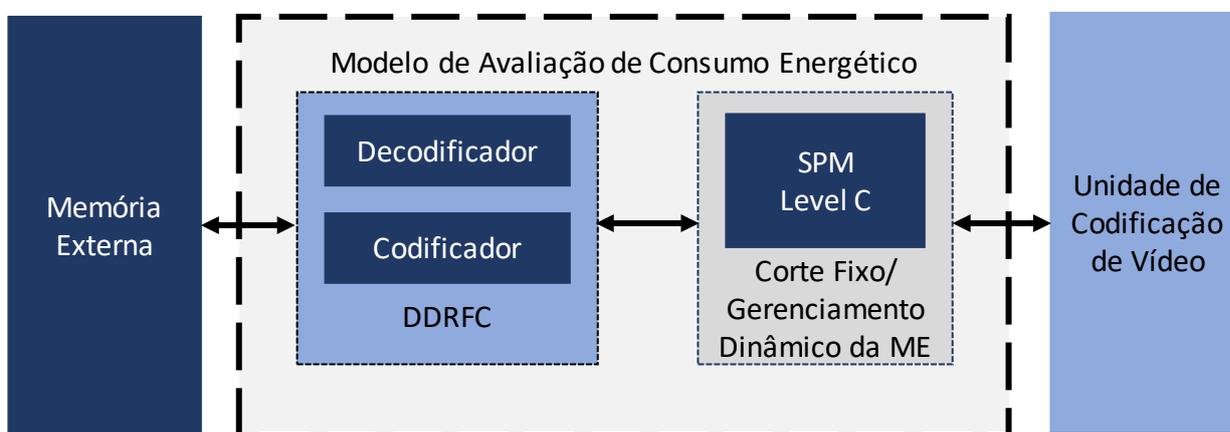


Figura 19. Sistema avaliado neste capítulo

está desabilitado e que a posição está acessível para o TZS. Deste modo, o TZS não consegue acessar blocos que possuem uma ou mais de suas amostras representadas por uma posição de valor '1' no *bitmap*, pois estes blocos necessitam de informações que ultrapassam os limites disponíveis para acesso na área com corte. Assim, um bloco fora dos limites disponíveis, esta comparação é ignorada e o TZS continua sua busca pelo melhor casamento.

Para avaliar o impacto na eficiência de codificação e verificar se o corte no formato diamante, como no *bitmap*, é mais eficaz do que um corte quadrado, foram realizados testes com o corte fixo de 33,35% e uma redução da área em formato quadrado, reduzindo a SR [-64, +63] para uma SR [-32, +31], uma opção suportada pelo HM 16.6. Estas duas reduções removem praticamente a mesma quantidade de área quando comparadas com a área de busca original (sem cortes), já que reduzir a área de busca original para uma SR [-32, +31] representa um corte de 33,34%. Uma maior compreensão da diferença entre os tamanhos da área de busca com SR [-64, +63] e SR [-32, +31] pode ser obtida por meio da Figura 21.

Na Figura 21, a imagem da esquerda mostra a área de busca com SR [-64, +63] e a imagem da direita representa a área de busca com SR [-32, +31] sobreposta a uma área de busca com SR [-64, +63]. As regiões em azul em ambas as imagens representam a área de busca total de cada configuração de tamanho e as setas representam os vetores de *search range*, que apontam para os limites máximos onde um bloco pode começar se formar (amostra referente ao canto superior esquerdo), durante a estimação de movimento. Desta forma, todas as setas partem do bloco central, que tem o tamanho máximo de uma CTU e neste trabalho é colocalizado ao bloco sendo codificado. Essas setas delimitam uma área de busca que compreende todo o bloco central e a região máxima a partir deste bloco onde a *search range* pode determinar o início de uma nova CTU. Tanto na Figura 21 quanto neste trabalho, este tamanho de CTU é 64x64. Desta forma, uma SR [-32, +31] e tamanho de bloco 64 representam uma área de busca de tamanho total igual a 128x128 amostras, já que $(2 \times SR_h + BS) \times (2 \times SR_v + BS) = (2 \times 32 + 64) \times (2 \times 32 + 64) = 128 \times 128$. A área de busca total com SR [-64, +63] tem dimensões totais de 192x192 amostras, já calculada anteriormente. Assim, a SR [-32, +31] implica em uma área total que corresponde a 66,66% de uma área de busca com SR [-64, +63].

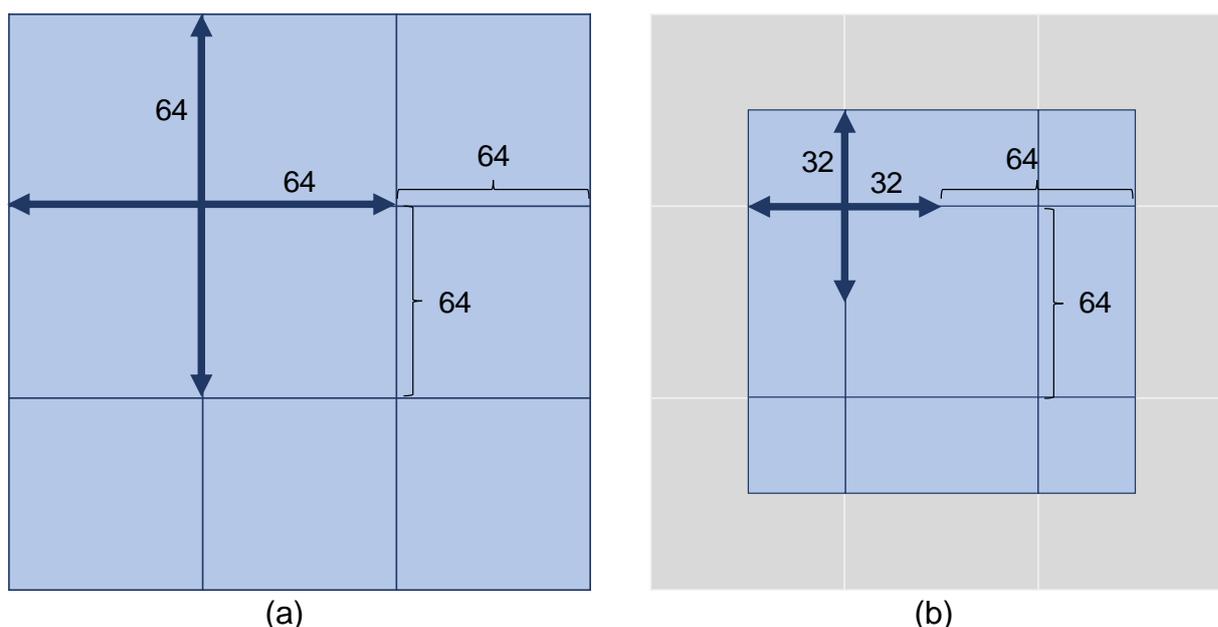


Figura 21. Relação entre as áreas de busca: (a) *search range* [-64, +63] e (b) *search range* [-32, +31]

Isto posto, os testes realizados aqui consideraram as seis sequências e as configurações da análise dos acessos utilizadas no capítulo anterior. A Tabela 1 contém os resultados de qualidade, medida em PSNR, e taxa de bits, ou *bitrate*, para os vídeos testados quando a área de busca original é utilizada e a diferença de qualidade e *bitrate* em relação aos resultados da área de busca original quando ambos os cortes discutidos até então são realizados. Conforme a Tabela 1, percebe-se, com os resultados médios, que o corte fixo 33,35% (CF 33,35%) da área de busca obteve pouca variação nos dois eixos avaliados quando comparado com a solução base. Em comparação com a SR [-64, +63], o corte fixo em formato diamante adotado apresentou resultados muito próximos de qualidade aliado a um aumento pequeno na taxa de bits. O SR [-32, +31] por outro lado, causou um aumento no significativo no *bitrate* em relação ao SR [-64, +63] e apresentou uma diferença mais notável na perda de qualidade dos vídeos. Estes resultados apontam que o corte em formato diamante é mais vantajoso do que o corte em formato quadrático, pois respeita os dados mais acessados e remove aqueles menos acessados, resultando em baixas variações na eficiência de codificação.

Tabela 1. Resultados de *bitrate* (bits/s) e PSNR (dB) para search range 64, 32 e corte fixo de 33,35%

Vídeo	QP	SR [-64,+63]		SR [-32,+31] (CF 33,34%)		CF 33,35%	
		<i>Bitrate</i>	PSNR	Δ <i>Bitrate</i>	Δ PSNR	Δ <i>Bitrate</i>	Δ PSNR
BasketballDrive	22	23412,75	39,4033	221,1400	0,0010	20,1560	0,0013
	27	8235,48	37,4981	90,4280	-0,0028	10,1680	0,0005
	32	3902,88	35,4356	50,1440	-0,0121	1,4920	-0,0043
	37	2004,41	33,1959	34,8600	-0,0196	2,7640	-0,0042
BQTerrace	22	67601,88	38,8586	43,7376	-0,0021	15,0624	0,0002
	27	14220,19	35,0889	3,7392	-0,0017	-1,7616	-0,0002
	32	3969,98	32,9138	9,6192	0,0002	1,7952	0,0006
	37	1302,40	30,7104	6,7200	-0,0045	-1,0800	0,0015
Cactus	22	26846,71	38,4122	53,1880	-0,0013	-14,9600	-0,0002
	27	7675,30	36,3302	24,5920	0,0001	-12,6080	0,0017
	32	3350,42	34,1431	20,1520	0,0015	3,7240	-0,0015
	37	1628,62	31,7946	12,1480	-0,0024	1,0520	0,0027
Kimono	22	7049,33	41,8403	-1,9373	-0,0027	1,2019	0,0009
	27	3430,57	39,8931	-1,4554	0,0034	0,0326	0,0005
	32	1692,53	37,1544	1,1079	-0,0021	0,7392	0,0003
	37	858,61	34,3299	-0,6221	-0,0106	-1,9718	-0,0064
ParkScene	22	10407,88	39,5999	11,6794	0,0001	-0,0691	0,0001
	27	4096,93	36,7013	7,4650	-0,0002	0,7699	0,0009
	32	1707,40	33,9097	5,4874	0,0017	-0,1306	0,0025
	37	738,33	31,3423	2,5882	0,0231	-0,5472	0,0128
Tennis	22	9916,66	40,7181	96,3628	-0,0041	4,6022	-0,0006
	27	4756,17	38,4683	60,7027	-0,0082	1,4477	-0,0014
	32	2348,28	35,8441	31,7817	-0,0257	2,9587	-0,0012
	37	1225,29	33,3676	19,4131	-0,0332	0,6221	-0,0017
Média	-	8849,12	36,1231	33,4601	-0,0043	1,4775	0,0002
Desvio Padrão	-	14235,12	3,1167	48,7166	0,0108	7,0355	0,0034

Ainda na Tabela 1, percebe-se que vídeos como as sequências *BasketballDrive*, *Cactus* e *Tennis*, alcançaram uma variação grande em *bitrate* quando a SR [-32, +31] foi adotada. Por outro lado, a sequência *Kimono* obteve uma variação baixa em taxa de bits, mas atingiu uma perda considerável em qualidade quando comparada com o CF. Isso ocorre porque os vídeos *BasketballDrive*, *Kimono* e *Tennis* possuem alta movimentação, assim como regiões do vídeo *Cactus*. Quando estas características são encontradas, é mais provável que o bloco de maior similaridade seja encontrado em regiões mais afastadas do centro da área de busca. Assim, a SR [-32, +31] pode não abranger estas regiões devido ao seu corte em formato quadrado. Quando o bloco de maior similaridade não é encontrado, outro bloco é escolhido pelo algoritmo de ME. Essa nova escolha pode gerar uma redução na eficiência de codificação, já que a escolha de um diferente vetor de movimento impacta na codificação do quadro atual e também nos próximos quadros, que terão o quadro atual como um quadro de referência. Desta forma, é importante que uma avaliação do impacto de soluções na eficiência de codificação seja feita, a fim de assegurar baixa variação em qualidade e na taxa de bits.

Como os resultados do CF 33,35% foram positivos, novos cortes em formato diamante foram adotados. Assim, foram escolhidas as regiões que correspondem a 75% e 55% dos dados mais acessados na área de busca. Desta forma, os cortes realizados removeram 65,60% (CF 65,60%) e 81,36% (CF 81,36%) da área de busca, respectivamente. Esses cortes estão ilustrados na Figura 22.

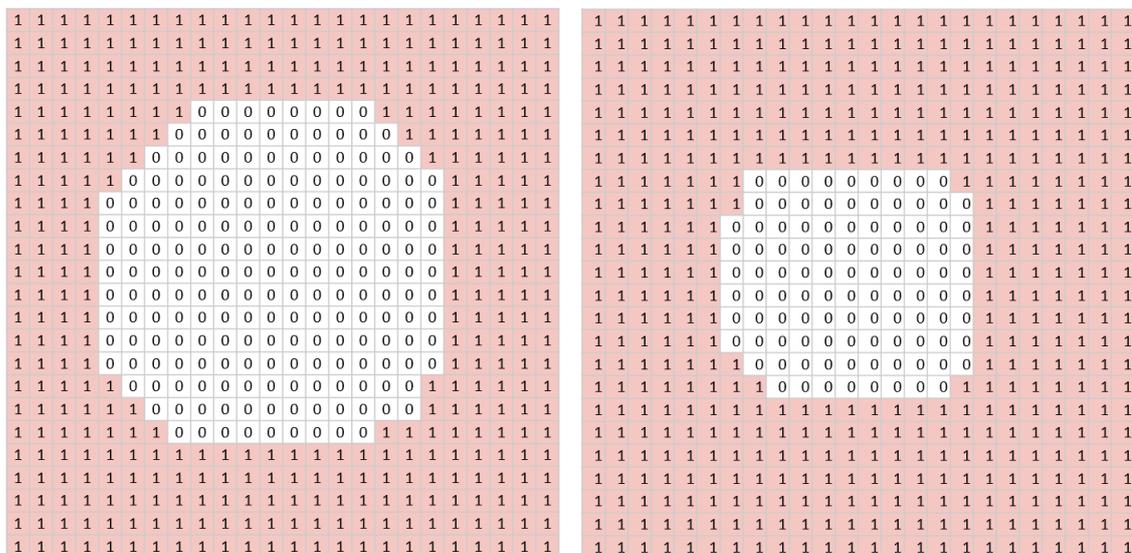


Figura 22. *Bitmaps* referentes aos cortes fixos: (a) corte fixo de 65,60% e (b) corte fixo de 81,36% da área de busca

Os resultados de diferença de *bitrate* e PSNR em relação a área de busca original (SR [-64, +63]) estão presentes na Tabela 2.

Tabela 2. Resultados de *bitrate* (bits/s) e PSNR (dB) para cortes fixos de 65,60% e 81,36% em relação à área de busca original

Vídeo	QP	CF 65,60%		CF 81,36%	
		Δ Bitrate	Δ PSNR	Δ Bitrate	Δ PSNR
BasketballDrive	22	72,8920	0,0023	248,5040	-0,0031
	27	29,1280	-0,0016	124,2800	-0,0153
	32	14,6080	-0,0048	72,1400	-0,0411
	37	15,4800	-0,0133	58,5960	-0,0764
BQTerrace	22	31,1904	-0,0009	24,7536	-0,0022
	27	3,2208	-0,0013	6,1008	-0,0031
	32	9,0048	-0,0010	9,0816	-0,0006
	37	-1,3152	-0,0066	0,4032	-0,0090
Cactus	22	49,3680	0,0006	66,5280	-0,0008
	27	11,8640	0,0001	33,1720	-0,0006
	32	7,2880	-0,0022	20,6720	-0,0051
	37	8,5560	-0,0023	19,4920	-0,0057
Kimono	22	-0,0442	-0,0011	19,7011	-0,0026
	27	-1,0003	0,0024	11,7542	-0,0049
	32	-0,2572	0,0052	8,2637	-0,0120
	37	-2,2099	-0,0176	7,4016	-0,0488
ParkScene	22	7,7530	-0,0003	15,8112	0,0010
	27	2,6573	0,0006	9,8055	-0,0016
	32	4,3968	0,0017	7,7683	-0,0058
	37	0,5587	0,0124	4,2471	0,0139
Tennis	22	26,7705	-0,0033	103,0694	-0,0074
	27	23,3030	-0,0036	76,9728	-0,0154
	32	14,2080	-0,0149	49,7049	-0,0387
	37	12,0960	-0,0201	41,2320	-0,0782
Média	-	14,1465	-0,0029	43,3106	-0,0151
Desvio Padrão	-	17,7943	0,0073	55,0402	0,0239

Pela Tabela 2, nota-se que o CF 65,60% da área de busca apresenta resultados melhores de eficiência de codificação do que os apresentados pela SR [-32, +31] na Tabela 1. Este corte tem uma menor perda em qualidade e um menor aumento da taxa de bits que a SR [-32, +31]. Isso ocorre porque quando a SR está configurada com o intervalo [-32, +31], a máquina do TZS somente terá esta área disponível para acesso. No entanto, como os cortes em formato diamante implementados neste trabalho são feitos sobre uma SR [-64, +63], o TZS consegue realizar um maior número de expansões, dado o seu formato de busca. Em alguns casos, estas expansões podem levar a acessos a blocos disponíveis dentro da área de busca em formato diamante, no caso da fase de Refinamento. Estes acessos, mais numerosos do que na SR [-32, +31] podem levar ao encontro de blocos que apresentem um melhor casamento.

Desta forma, estes resultados reafirmam as vantagens de se manter uma área de busca em formato diamante que recebe as requisições de acesso do TZS, mas permita que o acesso seja realizado apenas nas amostras presentes dentro da área de busca disponível. Por outro lado, quando comparado ao CF 33,35% realizado, o CF 65,60% atinge reduções maiores em qualidade e sofre um aumento considerável no *bitrate*. Analisando os resultados por vídeo, percebe-se que quanto mais agressivo o corte, maior o impacto na eficiência de codificação, independente da característica apresentada. Esse comportamento também se manteve no corte mais agressivo testado, o CF 81,36%, que foi a solução de corte fixo que atingiu o maior aumento em taxa de bits e a maior perda em qualidade.

Como há uma dificuldade em avaliar os impactos em qualidade causados por ganhos de compressão, a métrica BD-Rate vem sendo amplamente utilizada. O BD-Rate é uma métrica que relaciona a variação do *bitrate* para a mesma qualidade, medida em PSNR. Assim, os resultados de cada vídeo são comparados com os resultados base, obtidos com a SR [-64, +63] sem cortes. Os resultados da Tabela 3 confirmam as conclusões obtidas anteriormente, onde o CF 33,35% foi o que obteve os melhores resultados em eficiência de codificação. Este corte obteve um BD-Rate médio negativo e foi o que apresentou o menor BD-Rate para todos os vídeos. Um BD-Rate negativo indica que, para manter a mesma qualidade, o vídeo necessitou de uma taxa de bits menor do que a com a qual ele está sendo comparado. Neste caso, alguns vídeos de algumas soluções necessitariam de uma taxa de bits menor do que

Tabela 3. Resultados de BD-Rate para as variações de tamanho da área de busca

Vídeo	SR [-32,+31]	CF 33,35%	CF 65,60%	CF 81,36%
BasketballDrive	2,1526%	0,2083%	1,0799%	4,7787%
BQTerrace	0,5430%	-0,1190%	0,1673%	0,3640%
Cactus	0,6697%	-0,0424%	0,4400%	1,1027%
Kimono	0,1780%	-0,0528%	0,1902%	2,1037%
ParkScene	-0,4344%	-0,4908%	-0,3358%	0,1799%
Tennis	2,2938%	0,1013%	1,3196%	4,9770%
Média	0,9005%	-0,0659%	0,4769%	2,2510%
Desvio Padrão	0,2262%	3,9112%	0,4618%	0,2262%

a área de busca com SR [-64, +63] precisaria para transmitir o vídeo com a mesma qualidade.

Estes resultados podem ocorrer pelo fato de que o objetivo do TZS é encontrar o bloco mais similar ao que está sendo codificado no momento. Para isso, o algoritmo considera apenas o bloco com o menor resíduo (a menor diferença) em relação ao bloco que está sendo codificado e desconsidera outros detalhes que podem impactar no resultado final de toda a codificação, como o tamanho do vetor de movimento ou a similaridade do vetor de movimento de blocos vizinhos, por exemplo. Além disso, o CF 33,35% remove uma área equivalente a um terço da área de busca original. Assim, vetores de movimento menores, aliados a blocos com uma similaridade muito próxima, podem gerar taxas de bits menores e uma variação muito baixa em PSNR, levando a resultados negativos em BD-Rate.

Ainda na Tabela 3, fica claro que conforme o corte na área de busca vai se tornando mais agressivo, o impacto na eficiência de codificação vai se tornando maior. Enquanto o CF 33,35% obteve os resultados mais próximos à área de busca sem cortes, o CF 82,64% mostrou que vídeos de alta movimentação como *BasketballDrive* e *Tennis* são os mais prejudicados em termos de codificação, atingindo um aumento superior a 4,7% no BD-Rate.

Além da eficiência de codificação, este trabalho traz também uma preocupação relacionada com a eficiência energética de codificadores de vídeo. Como o consumo de energia dos codificadores de vídeo está diretamente relacionado com o número de acessos feitos na memória externa durante a etapa de estimação de movimento, um

Tabela 4. Total de amostras acessadas pelos tamanhos de áreas de busca

Vídeo	SR [-64, +63] (x10 ¹¹)	SR [-32, +31] (x10 ¹¹)	CF 33,35% (x10 ¹¹)	CF 65,60% (x10 ¹¹)	CF 81,36% (x10 ¹¹)
BasketballDrive	3,63	1,48	2,76	1,75	1,01
BQTerrace	0,91	0,59	0,79	0,63	0,47
Cactus	0,74	0,47	0,65	0,52	0,39
Kimono	2,64	1,26	2,13	1,49	0,94
ParkScene	0,82	0,51	0,70	0,55	0,41
Tennis	3,54	1,41	2,70	1,70	1,01
Média	2,05	0,95	1,62	1,11	0,71
Desvio Padrão	1,39	0,48	1,02	0,60	0,31

menor número de blocos acessados durante a ME, gera um menor custo de energia. Deste modo, a Tabela 4 apresenta os resultados do total de amostras acessadas pelo algoritmo TZS na área de busca durante a codificação de cada um dos vídeos testados. A redução do total de amostras acessadas atingida pelas áreas de busca com corte quando comparadas com a área de busca original, encontra-se na Tabela 5.

A partir da Tabela 4 e da Tabela 5, nota-se que a maior redução no total de amostras acessadas foi atingida com o uso do corte mais agressivo. Isto é esperado dado que a área de busca onde o TZS pode realizar a busca pelo melhor casamento

Tabela 5. Redução do total de amostras acessadas

Vídeo	SR [-32, +31]	CF 33,35%	CF 65,60%	CF 81,36%
BasketballDrive	59,11%	24,12%	51,70%	72,19%
BQTerrace	35,13%	12,71%	30,03%	47,98%
Cactus	36,43%	11,90%	29,42%	46,84%
Kimono	52,18%	19,19%	43,55%	64,55%
ParkScene	37,64%	14,66%	32,86%	50,13%
Tennis	60,28%	23,82%	52,04%	71,51%
Média	46,79%	17,73%	39,94%	58,87%
Desvio Padrão	11,74%	5,45%	10,55%	11,91%

é menor. No entanto, entre as áreas de SR [-32,+31] e o CF 33,35%, que correspondem a áreas praticamente equivalentes, o corte quadrado teve uma redução consideravelmente maior no número de amostras acessadas do que a área em formato diamante. Isto se deve ao fato de que, ao fazer o corte quadrado, áreas que são muito acessadas ficam de fora da área de busca resultante, causando uma redução maior no número de amostras acessadas, mas também um impacto negativo na eficiência de codificação, como claramente observado nos resultados apresentados pelos vídeos *BasketballDrive*, *Kimono* e *Tennis*, tanto em BD-Rate quanto em número de acessos. Estes vídeos, por apresentarem maior movimentação, precisam expandir a busca do TZS em maior escala do que os demais vídeos. Com áreas de busca menores ou quadradas, pode ser que o encontro do melhor casamento esteja fora da área de busca disponível para acessos do TZS. Assim, áreas maiores e em formato diamante são beneficiadas por permitirem um maior número de expansões do algoritmo. Além disto, nestas áreas que permitem um maior número de expansões, há uma maior incidência de etapas de Busca *Raster*, já que a constante *iRaster* possui um valor *default* pequeno. Deste modo, estes vídeos de alta movimentação demonstraram pouca redução no acesso aos dados com o CF 33,35%, mas obtiveram os melhores resultados em qualidade e *bitrate* com este mesmo corte.

Por outro lado, alguns vídeos não necessitam de um grande número de expansões durante a busca na ME, pois tendem a apresentar baixa movimentação ou baixa ocorrência de texturas. Assim, vídeos como *BQTerrace*, *Cactus* e *ParkScene* atingem uma menor redução no número de dados acessados quando cortes fixos foram realizados. No caso destes vídeos, como a probabilidade do encontro do maior casamento está próximo ao centro da área de busca, há uma maior chance de o melhor casamento ser encontrado na Busca Inicial sem uma grande incidência de expansões. Assim, a Busca *Raster* acaba sendo menos utilizada e a redução da área de busca disponível não traz grandes impactos na qualidade e taxa de bits desses vídeos.

Finalmente, embora a redução no número de amostras acessadas tenha sido maior nos casos com SR [-32,+31] e com CF 81,36%, estas configurações de tamanho de área de busca foram as que apresentaram os maiores impactos negativos na eficiência de codificação quando adotadas.

4.1 Projeto da memória de busca

Os resultados alcançados com os cortes fixos na área de busca da ME apontam que vídeos de baixa movimentação ou textura não sofrem grande prejuízo em termos de eficiência de codificação quando são adotados cortes na janela de busca. Ou seja, esses vídeos conseguem se comportar bem quando áreas de busca menores são adotadas. Os vídeos *BQTerrace* e *ParkScene*, por exemplo, obtiveram variações pequenas em BD-Rate em todos os experimentos com corte fixo. Por outro lado, vídeos de alta movimentação ou textura alcançaram variações mais altas em BD-Rate, que foram se tornando mais notáveis ao passo em que o corte de regiões foi se tornando mais agressivo, como nos casos dos vídeos *BasketballDrive* e *Tennis*.

Como vídeos de alta movimentação ou textura exigem um maior esforço computacional (maior número de comparações) na ME para encontrar o bloco mais similar, a redução na área de busca provocada pelo uso dos cortes fixos implica diretamente na diminuição do número de amostras acessadas. No caso desses vídeos, as reduções alcançadas em número de blocos comparados se tornam mais pronunciadas do que nos casos dos vídeos de baixa movimentação ou textura. Isso ocorre porque vídeos de baixa movimentação ou textura usam muito pouco as regiões mais próximas às bordas da área de busca e, portanto, ao remover essas regiões, o impacto em número de comparações é menor.

Desta forma, vídeos diferentes necessitam de esforços computacionais distintos e se beneficiam de formas diferentes quando os cortes fixos são adotados. Para melhor explorar a relação entre eficiência de codificação e consumo de energia, altamente influenciado pelo número de amostras comparadas, este trabalho propõe um projeto de memória interna e um gerenciador dinâmico da estimação de movimento a partir desta memória. Este sistema é composto por uma SPM, que permite *power gating* a nível de setores. Desta forma, o gerenciador proposto desliga setores de memória quando os acessos realizados neles se tornam irrelevantes para a ME. Com o desligamento de setores, além da redução de consumo estático na memória interna, proveniente do número de células SRAM desligadas, também se atinge uma redução no consumo dinâmico do sistema como um todo, já que as regiões desligadas implicam em um menor número de blocos disponíveis para acesso durante a ME. Assim, dados indisponíveis na memória não serão entregues para a ME, mesmo que sejam requisitados, reduzindo o acesso à memória. Este sistema (projeto de

Tabela 6. Média dos acessos por setor

Vídeo	Setor 0	Setor 1	Setor 2	Setor 3
BasketballDrive	50,84%	19,41%	19,82%	9,93%
BQTerrace	84,55%	7,87%	5,47%	2,10%
Cactus	84,23%	7,47%	6,00%	2,30%
Kimono	57,78%	18,65%	16,76%	6,80%
ParkScene	85,14%	7,08%	5,43%	2,36%
Tennis	48,87%	20,63%	20,98%	9,53%
Média	68,57%	13,52%	12,41%	5,50%
Desvio Padrão	17,86%	6,66%	7,55%	3,72%

A relação dos acessos por setor se encontra na Tabela 6. Por meio desta tabela, nota-se que a maioria dos acessos para todos os vídeos é feita no Setor 0, de onde parte a busca do TZS durante a fase de Busca Inicial. Porém, vídeos de baixa movimentação ou textura foram os que apresentaram o maior número de acessos realizados no Setor 0. Para estes vídeos, mais de 80% dos acessos foram realizados neste setor. Vídeos de alta movimentação tiveram 50% dos acessos médios em torno do Setor 0 e cerca de 20% em ambos Setor 1 e Setor 2. O Setor 3 foi o que obteve o menor número de acessos. Em vídeos de baixa movimentação, o número de acessos ficou abaixo dos 3% e para vídeos de alta movimentação, este número não chegou aos 10%. Dado que o Setor 3 foi o setor com o menor volume de acessos e que o CF 33,35% (união dos setores 0, 1 e 2) apresentou uma baixa variação em eficiência de codificação, com uma redução superior a 20% no número de amostras acessadas para alguns vídeos, o Setor 3 foi desconsiderado do próximo estudo e a área correspondente foi removida do projeto da memória de busca.

A Tabela 7 contém o total de CTUs, em porcentagem, durante a codificação de 100 quadros de cada vídeo, que tiveram como o mais relevante o Setor 0, o Setor 1 e o Setor 2. Para determinar o setor mais relevante solicitado pela CTU em sua codificação, foram utilizadas as médias encontradas na Tabela 6. Desta forma, se o Setor 2 recebeu uma quantidade de acessos igual ou superior a 12,41% do total (conforme a Tabela 6), então ele foi o setor mais relevante na codificação da CTU. Caso contrário, se o Setor 1 recebeu 13,52% do total de acessos ou mais, então o Setor 1 foi o setor mais relevante requisitado. Por fim, caso o número de acessos nos

Tabela 7. Ocorrência dos setores mais relevantes em número de CTUs (%)

Vídeo	Setor 0	Setor 1	Setor 2
BasketballDrive	15,86%	3,00%	81,13%
BQTerrace	77,44%	7,64%	14,92%
Cactus	71,53%	7,71%	20,76%
Kimono	18,49%	12,44%	69,07%
ParkScene	76,38%	5,68%	17,94%
Tennis	14,08%	2,86%	83,07%
Média	45,63%	6,55%	47,81%
Desvio Padrão	32,39%	3,58%	33,20%

setores 1 e 2 não atendam a essas quantidades, determina-se que o Setor 0 foi o setor mais relevante. Por meio desta tabela, é notável que vídeos de alta movimentação tiveram a grande maioria de suas CTUs, cerca de 80%, com o Setor 2 como o setor mais relevante. Vídeos de baixa movimentação, ao contrário, tiveram a maioria de suas CTUs, cerca de 77%, com Setor 0 como o setor mais relevante. O Setor 1 foi o setor que se apresentou menos vezes como o setor mais relevante em número de acessos por CTU. Isso se deve à forma como o TZS expande suas buscas nas fases de Busca Inicial, sempre com um distanciamento em potência de dois e fazendo três expansões acima do nível onde o melhor bloco atual foi encontrado. Assim, ao expandir a Busca Inicial fora do Setor 0, é muito provável que o Setor 2 seja acessado.

Como as ocorrências dos setores 0 e 2 como os setores mais relevantes em número de CTUs foram muito superiores às ocorrências do Setor 1, a fatia correspondente ao Setor 1 foi integrada ao Setor 2 e o mapa de setores da Figura 24 foi adotado no restante deste trabalho. Neste mapa, existem apenas dois setores, o Setor *I* (*inner*, interno) e o Setor *O* (*outer*, externo).

Este modelo de memória é o mesmo utilizado para as soluções com CF 33,35%, com o CF 81,13% e com gerenciamento dinâmico de ME. Enquanto no CF 33,35% os dois setores *O* e *I* ficam sempre ativos, no CF 81,36% apenas o Setor *I* fica ligado. No gerenciamento dinâmico, por outro lado, enquanto o Setor *I* está sempre ligado na memória interna, por ter tamanho equivalente à menor área de busca avaliada (CF 81,36%), o Setor *O* pode ser ligado ou desligado conforme o número de

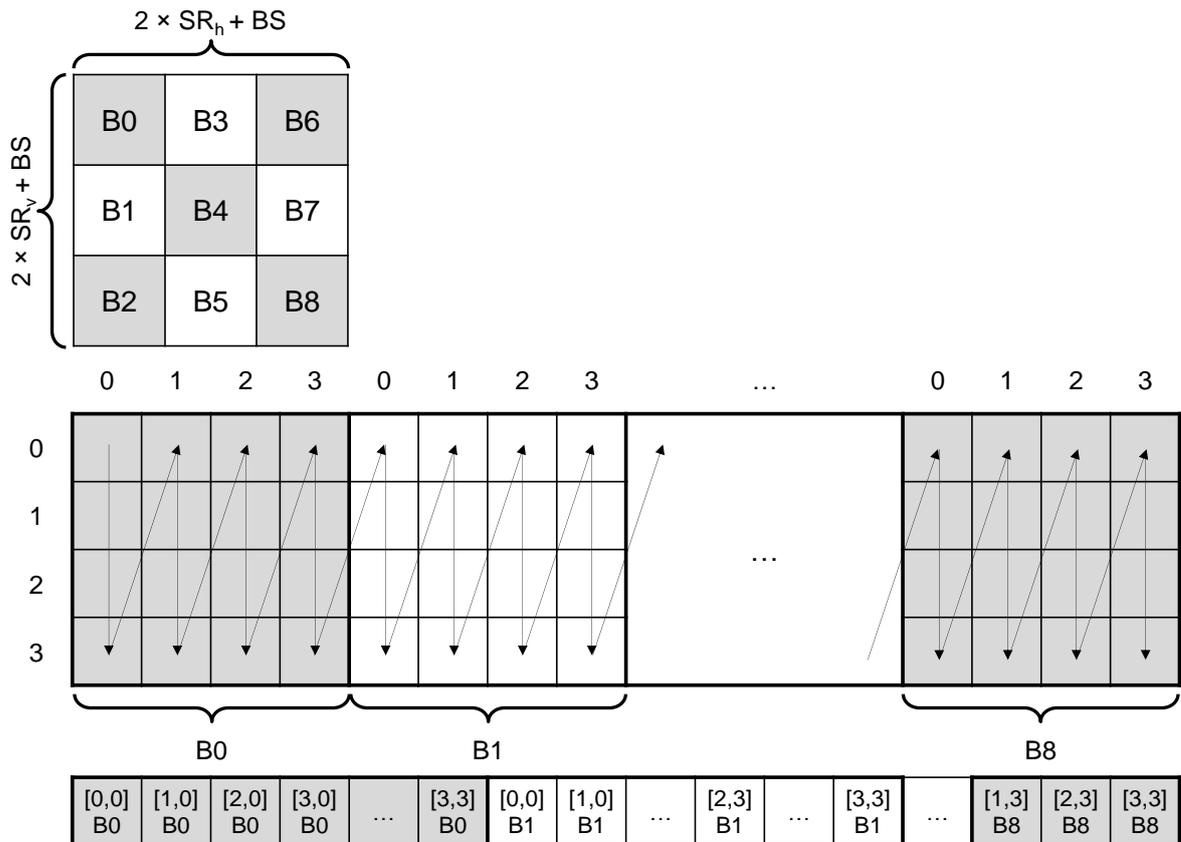


Figura 25. Organização dos dados no *Level C* (AMARAL, 2015)

de referência, ao contrário de granularidades menores. Assim, esta SPM física tem um tamanho equivalente a 504 bancos de memória, como o mapa de 21x24 posições. No entanto, os bancos de memória ativos neste SPM somam, no máximo, 384 bancos de memória (24576 bytes), de acordo com o CF 33,35% e, no mínimo, 103 bancos (6592 bytes), de acordo com o CF 81,36%. Estes bancos de memória podem ser ligados ou desligados conforme a requisição do gerenciador dinâmico da ME. Esta SPM foi implementada como um *buffer* circular onde o início da área de busca é indicado por uma *flag*. Isso permite que o *Level C* funcione de forma otimizada, evitando operações de escrita desnecessárias cada vez que um deslocamento de área de busca acontece. A organização lógica da memória interna se dá conforme a Figura 25.

O primeiro bloco da Figura 25 representa a área de busca com *search range* [-64, +63], conforme a organização lógica da memória implementada, que é preenchida na ordem B0, B1, ..., B8, onde B é um bloco de 64x64 amostras e cada B[x,y] tem o tamanho equivalente a um bloco de 8x8 amostras (64 bytes). Nesta figura, o cada bloco B se divide em 4x4 blocos de memória, sendo utilizado apenas para

ilustração, onde cada bloco $B[x,y]$ armazenaria então o equivalente a um bloco de 16×16 amostras (256 bytes). Além disso, cada um dos blocos B é armazenado neste *buffer* coluna a coluna, como pode ser visto na Figura 25. No entanto, como a área de busca com cortes não é quadrada, os bancos de memória da SPM precisam ser ligados ou desligados conforme a transição de CTUs e dos setores solicitados para a codificação destas CTUs. Assim, considerando o reuso de dados e uma transição de três CTUs vizinhas A , B e C , onde o Setor O se mantém ligado durante todo o processamento destes blocos (caso $O-O-O$), o ligamento e desligamento das células de memória é ilustrado na Figura 26.

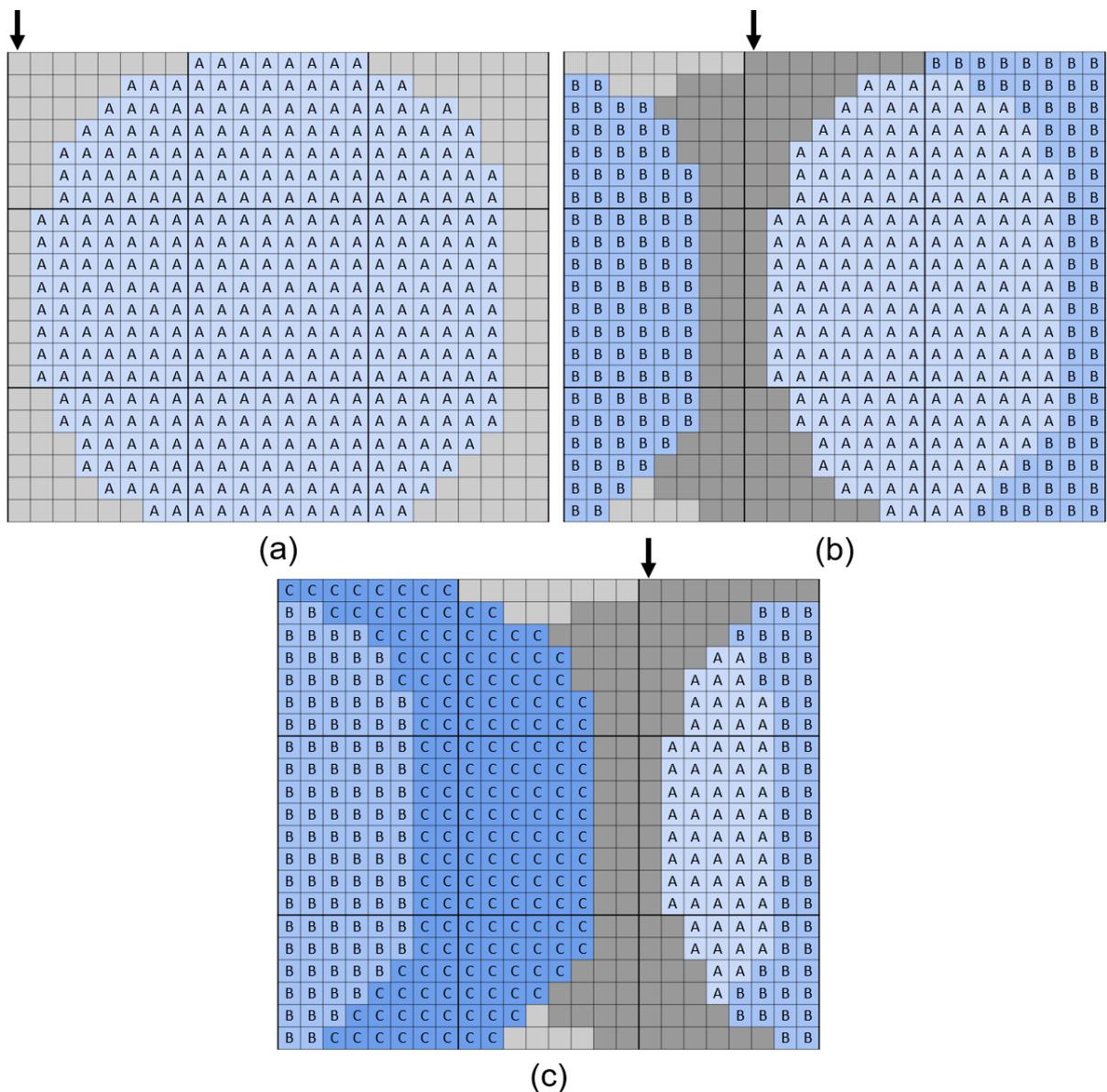


Figura 26. Comportamento das células SRAM no caso $O-O-O$: (a) área de busca da CTU A , (b) área de busca da CTU B e (c) área de busca da CTU C

Na Figura 26, os blocos com a letra A na primeira área de busca, que se inicia na primeira posição do mapa (indicada com a flecha), indicam os dados que foram lidos da memória externa, decodificados e armazenados na memória interna para a codificação da CTU A e regiões desligadas da memória interna (área cinza). A segunda área de busca, para a codificação da CTU B, começa na nona posição deslocada à direita do mapa da memória interna, onde estão armazenados os dados que iniciam a área de busca necessária. Neste caso, alguns dados trazidos da memória externa para a codificação da CTU anterior (CTU A) são desligados da memória interna (áreas em cinza escuro), alguns são mantidos e novos dados, que complementam a área de busca referente à CTU B são armazenados na memória interna. Esses dados são ilustrados com a letra B na figura. Por fim, na terceira área de busca, constam os dados necessários para a formação da área de busca da CTU C. Neste terceiro caso, os dados que iniciam a área de busca são indicados pela *flag* na décima sétima coluna do mapa da memória interna. Para formar a área de busca correspondente, os dados necessários e já armazenados para a codificação das CTUs A e B se mantêm ligados e novos dados são armazenados para completar a área de busca (indicados com a letra C).

Por fim, este projeto de memória de busca permite o controle da memória em nível de setores, possibilitando o *power gating* das células correspondentes a um determinado setor. Isto permite que haja uma redução energética estática, proveniente do desligamento das células SRAM, e também uma redução energética dinâmica, com a redução do número de amostras comparadas durante a codificação. No entanto, para que este gerenciamento possa atingir uma boa relação entre redução de consumo de energia e impacto em eficiência de codificação, é necessário que um controle eficiente seja adotado. Para isto, diferentes estratégias de gerenciamento dinâmico da estimação de movimento centrado na memória são abordadas no próximo capítulo.

5 GERENCIAMENTO DINÂMICO DA ESTIMAÇÃO DE MOVIMENTO

Neste capítulo serão abordadas estratégias para gerenciamento dinâmico da estimação de movimento. O projeto do gerenciador proposto neste trabalho pode assumir qualquer uma destas estratégias, baseado na relação entre qualidade de vídeo e redução do consumo de energia que for mais adequada. Assim, o sistema avaliado neste capítulo conta com uma memória de busca que possui um gerenciamento dinâmico, visando a ativação ou desativação do Setor O quando conveniente, visando sempre o menor consumo com o menor impacto possível na qualidade. Este sistema também é avaliado com o modelo de avaliação de consumo de energia proposto em (AMARAL, 2015), conforme a Figura 19, no capítulo 4.

As estratégias de gerenciamento dinâmico da ME apresentadas neste capítulo consideram diferentes granularidades: divisão do quadro em quadrantes, em linhas e em CTUs. No entanto, todas essas estratégias fazem suas decisões, de ativar ou não o Setor O, conforme uma matriz de referência que representa as solicitações de cada CTU de um quadro. Assim, um vídeo com resolução HD 1080p apresenta uma matriz de referência de 510 posições, já que um vídeo de 1920x1080 pixels contém 17 linhas e 30 colunas de CTUs de tamanho 64x64. Um exemplo de matriz de referência é encontrado na Figura 28. Esta matriz de referência é formada pelas solicitações das CTUs durante a codificação do primeiro quadro inter (Quadro 1) do vídeo *BQTerrace*.

Cada posição na matriz de referência é atualizada de acordo com o número de solicitações do TZS durante a codificação da CTU correspondente. Assim, uma posição na matriz pode receber o valor '1' quando a CTU correspondente solicitar a ativação do Setor O ou o valor '0', quando a CTU solicitar o desligamento do Setor O. Com o estudo do capítulo anterior, foi estabelecido que uma CTU solicita a ativação do Setor O quando um terço ou mais dos acessos requisitados durante a sua codificação são feitos fora do Setor I. Esta metodologia é seguida durante toda a codificação de um vídeo, de forma que a referência indicando se o Setor O foi solicitado ou não é sempre atualizada seguindo a codificação da CTU colocalizada do quadro anterior.

então, no total de solicitações do quadrante colocalizado no quadro anterior. Assim, na solução GD 4Q 1/5, se um quinto ou mais do total de CTUs de determinado quadrante solicitar o Setor O ativo, todas as CTUs deste quadrante receberão o Setor O ativo no quadro atual. Caso contrário, estas CTUs serão codificadas apenas com o Setor I. Na solução GD 4Q 1/3, a quantidade de CTUs necessária para a tomada de decisão aumenta para um terço ou mais do total de CTUs em um quadrante. O setor ativo para cada CTU na codificação do Quadro 2 do vídeo *BQTerrace* com o GD 4Q 1/3 se dá conforme a Figura 29, onde as CTUs que serão codificadas com o Setor O ativo são indicadas com 'O' e as que serão codificadas apenas com o Setor I são indicadas com 'I'. De acordo com esta figura, apenas o primeiro quadrante do quadro recebeu o Setor O ativo.

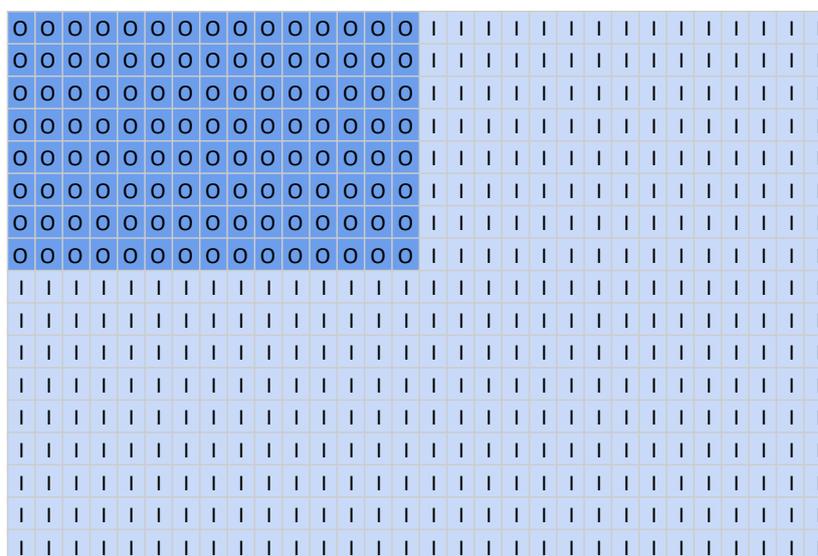


Figura 29. Setores ativos na codificação do Quadro 2 do vídeo *BQTerrace* com a solução GD 4Q 1/3

5.2 Gerenciamento por linha (GD L 1/2 e GD L 1/3)

O gerenciamento dinâmico por linha se dá conforme a quantidade de CTUs que solicitaram o Setor O ativo na codificação da linha colocalizada no quadro anterior. Assim, na solução GD L 1/3, se um terço ou mais do total de CTUs de uma determinada linha no quadro passado solicitou o Setor O ativo, esta mesma linha no quadro atual receberá o Setor O ativo. Na solução GD L 1/2, a quantidade de CTUs em uma linha que solicita o Setor O precisa ser igual a metade ou mais da quantidade

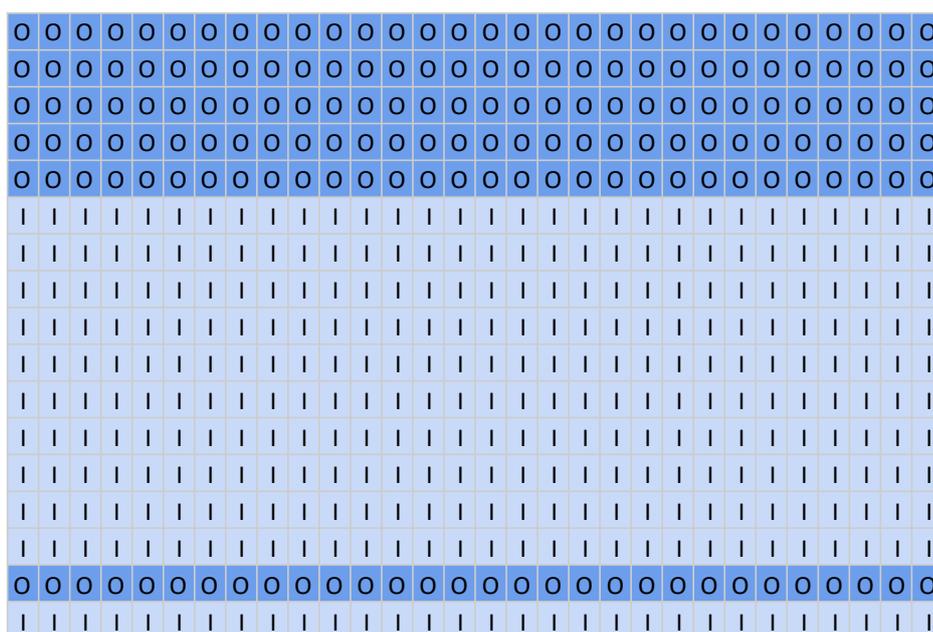


Figura 30. Setores ativos na codificação do Quadro 2 do vídeo *BQTerrace* com a solução GD L 1/3

total de CTUs em uma linha para que o Setor O permaneça ativo durante toda a codificação desta linha no próximo quadro. A Figura 30 ilustra o setor ativo na codificação de cada CTU do Quadro 2 com o GD L 1/3.

5.3 Gerenciamento por meia linha (GD ML 1/3)

O gerenciamento dinâmico por meia linha divide cada linha de CTUs do quadro em duas. Assim, se um terço das CTUs de uma meia linha colocalizada solicitar o Setor O ativo, a meia linha atual receberá o Setor O ativo durante a codificação de todas as suas CTUs. Caso contrário, toda a meia linha atual passará pela estimação de movimento com o Setor O inativo. O Quadro 2 é codificado conforme os setores ativos na Figura 31, quando essa solução é adotada.

5.4 Gerenciamento por CTU (GD CTU e GD CTU OT)

O gerenciamento dinâmico por CTU é o esquema com a menor granularidade. Nesse caso, cada CTU do quadro atual é codificada com o setor requisitado pela CTU colocalizada no quadro anterior. Porém, nesta solução é importante observar que um mesmo dado pode ser lido da memória externa e armazenado na memória interna mais de uma vez. Esse *overhead* na leitura de dados ocorre na transição entre os setores O e I quando, considerando três CTUs vizinhas e o reuso de dados, a primeira

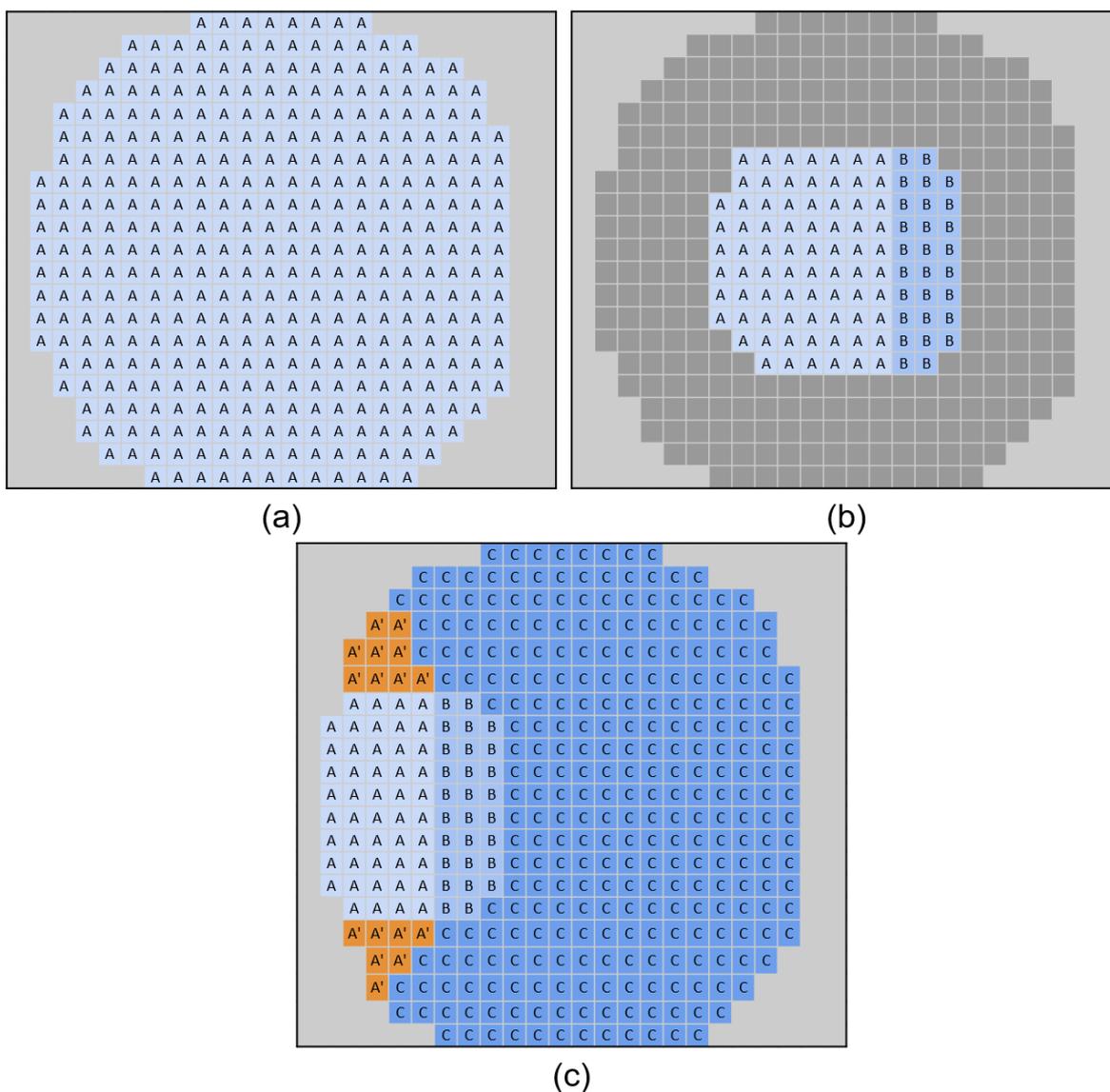


Figura 32. Ocorrência de overhead (caso O-I-O): (a) área de busca da CTU A, (b) área de busca da CTU B e (c) área de busca da CTU C

da CTU A, com o Setor O ativo, na transição para a codificação da CTU B, com o Setor O inativo, como demonstrado na Figura 33. Nesta figura, os dados armazenados na codificação das CTUs A, B e C são novamente indicados com a letra correspondente. No entanto, ao armazenar os dados faltantes para completar a área de busca necessária na codificação da CTU B, que receberia apenas o Setor I ativo, os dados armazenados anteriormente na codificação da CTU A não são desligados. Assim, quando a CTU C for codificada, os dados em comum das áreas de busca das CTUs A e C ainda estarão armazenados e a leitura duplicada destes dados na memória externa será evitada.

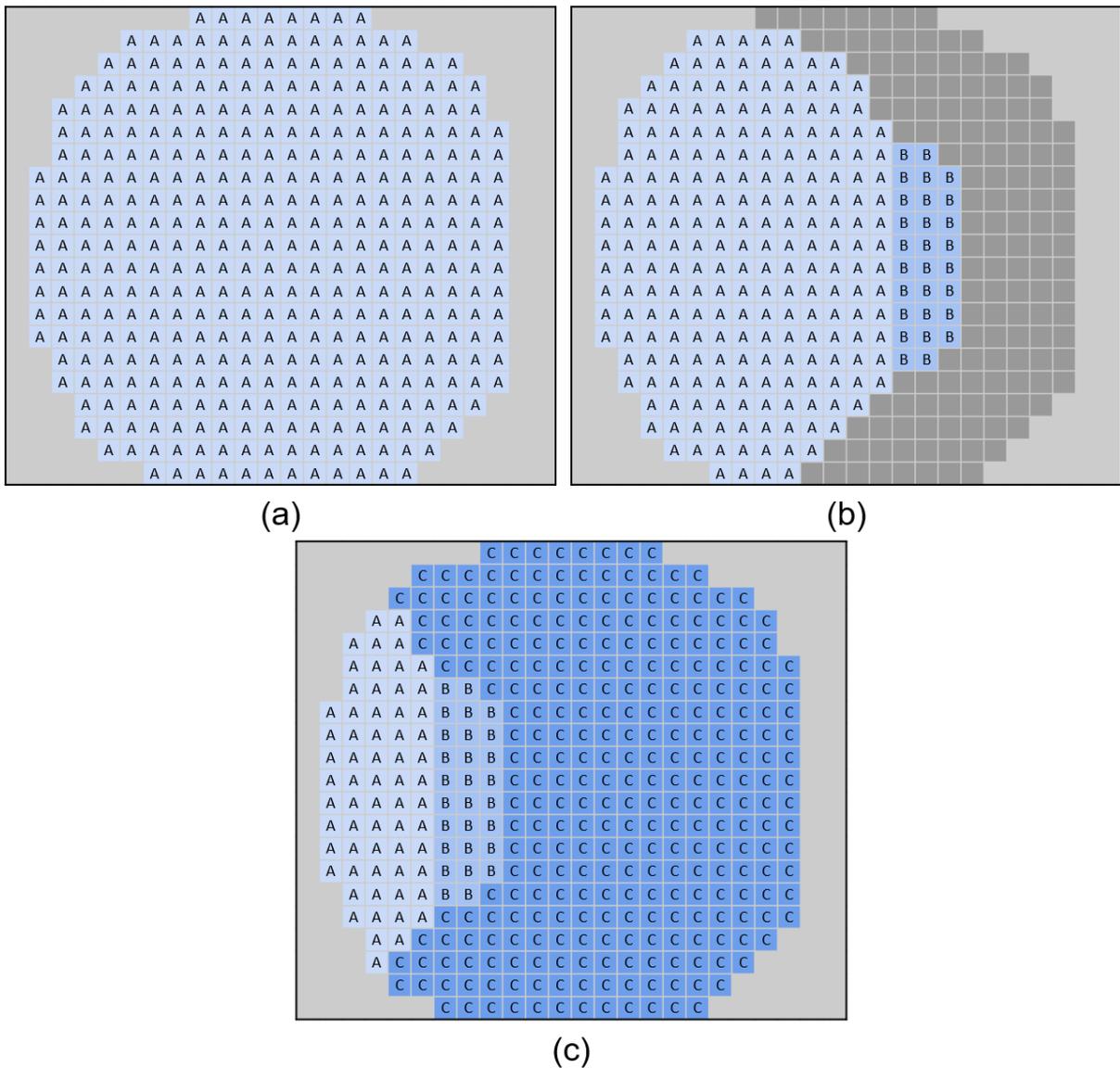
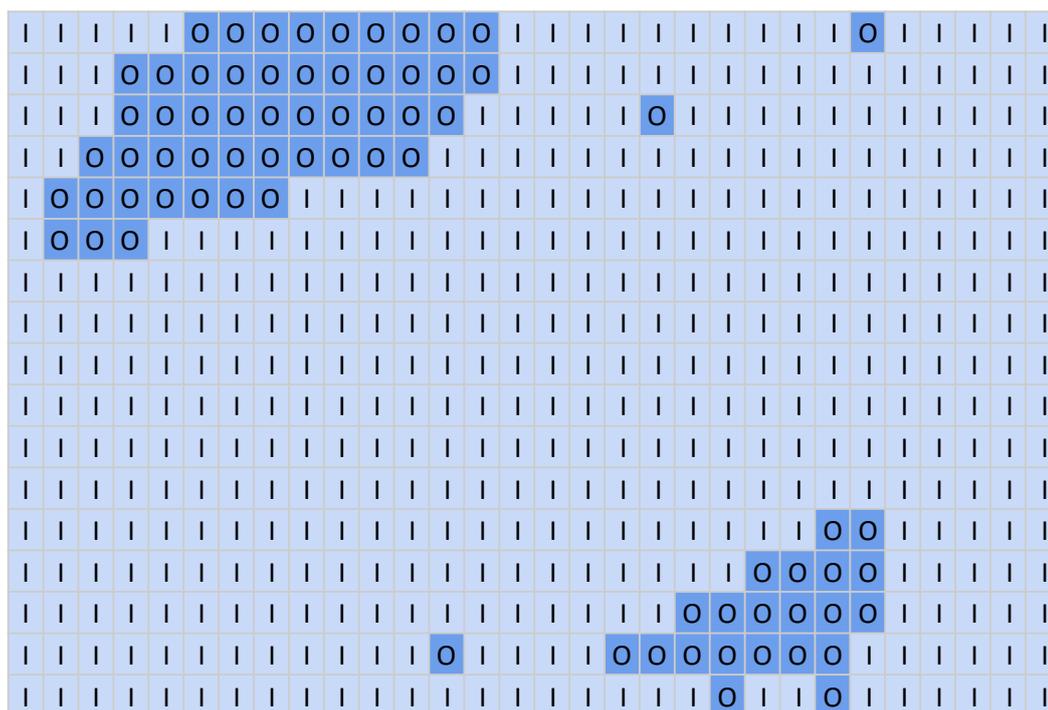


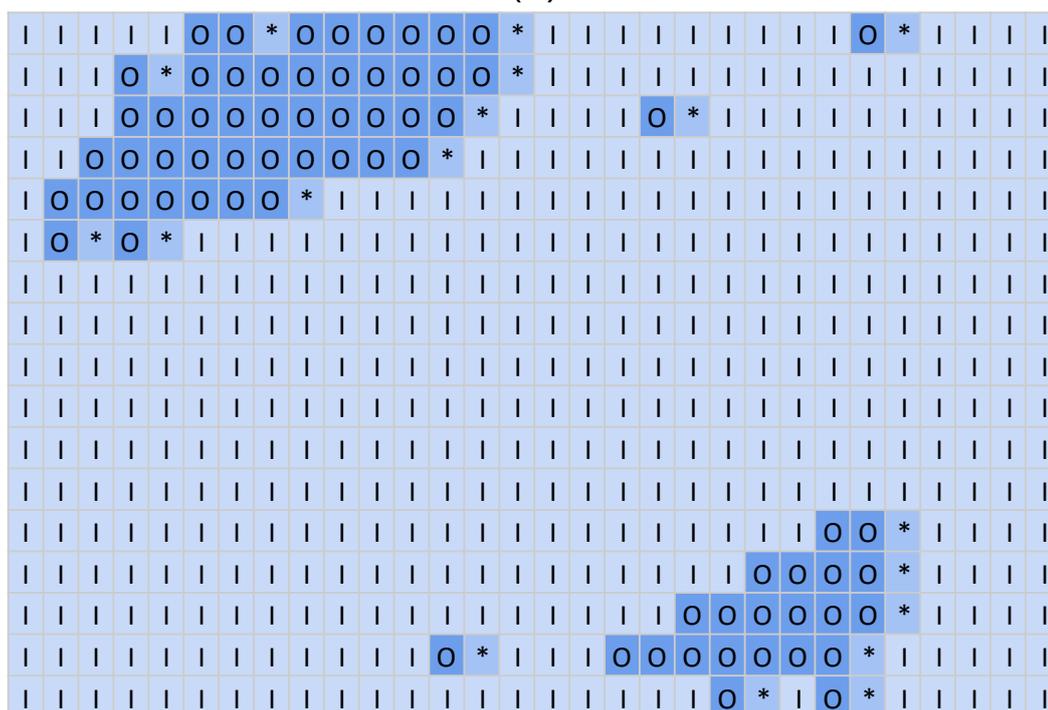
Figura 33. Solução otimizada para evitar o overhead no caso O-I-O: (a) área de busca da CTU A, (b) área de busca da CTU B e (c) área de busca da CTU C

Essa segunda solução de contornar o *overhead* permitiu que o gerenciamento dinâmico evoluísse para avaliar soluções que envolvem decisões tomadas sobre as solicitações das CTUs do quadro anterior e CTUs do mesmo quadro que já foram previamente codificadas. Com isso, quatro novos tipos de gerenciamento dinâmico da ME foram avaliados, onde os dados armazenados na memória interna quando o Setor O está ativo não são desligados na codificação da próxima CTU.

O setor ativo na codificação de cada CTU do Quadro 2 para as soluções GD CTU e GD CTU OT se encontram na Figura 34. No caso do GD CTU OT, a decisão de não desligar as células SRAM da CTU anterior é ilustrada com um “*” na Figura 34(b).



(a)



(b)

Figura 34. Setores ativos na codificação do Quadro 2 do vídeo *BQTerrace* com gerenciamento dinâmico: (a) solução GD CTU e (b) solução GD CTU OT

5.5 Gerenciamento pela maioria dos vizinhos (GD MV)

No gerenciamento dinâmico baseado na maioria dos vizinhos, a matriz de referência é consultada para cada CTU sendo codificada. Se a CTU colocalizada solicitar o Setor O ativo, a CTU sendo codificada terá o Setor O ativo em sua codificação. Caso contrário, se a maioria das oito posições vizinhas à CTU atual solicitarem o Setor O ativo, esta CTU terá o Setor O habilitado em sua codificação.

Como a matriz de referência é atualizada sempre após a codificação da CTU atual, o gerenciamento baseado na maioria dos vizinhos considera CTUs vizinhas que já foram codificadas no quadro atual (Quadro 2) e CTUs codificadas no quadro anterior (Quadro 1). Na Figura 35 há uma representação de como a decisão de ativar ou não o Setor O é tomada. Nesta figura, a cor cinza indica CTUs já codificadas no quadro atual, a cor branca indica CTUs que ainda não foram codificadas no quadro atual e a cor azul indica a próxima CTU a ser codificada. Assim, na matriz de referência (Figura 35(a)) é analisado se a maioria das CTUs vizinhas solicitou o Setor O ativo. Neste caso, o resultado foi positivo e a CTU foi codificada com o Setor O ativo, conforme a matriz de setores ativados da Figura 35(b). Só após a codificação da CTU atual, a matriz de referência na posição correspondente será atualizada com a informação do setor requisitado.

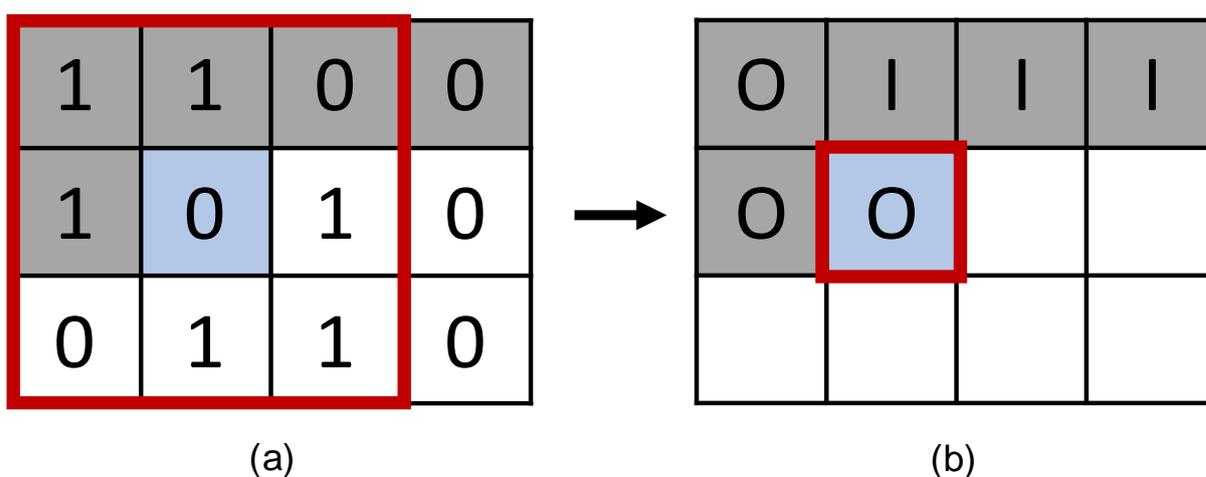


Figura 35. Ilustração da solução GD MV: (a) matriz de referência e (b) setores ativados

5.6 Gerenciamento pela expansão do Setor O (GD ESO)

A estratégia baseada na expansão do Setor O verifica se a CTU vizinha anterior a que está sendo codificada solicitou o Setor O ativo. Se sim, a CTU atual recebe o Setor O ativo. Caso contrário, é verificado se a CTU colocalizada no quadro anterior solicitou o Setor O ativo. Se sim, a CTU atual recebe o Setor O ativo. Se não, a CTU atual é codificada apenas com o Setor I ativo. Esta decisão é ilustrada na Figura 36, onde é necessário verificar se CTU imediatamente anterior e já codificada no quadro atual solicitou o Setor O ativo. Caso o Setor O não tenha sido solicitado, uma verificação da solicitação da CTU colocalizada no quadro anterior é feita. Como a CTU imediatamente anterior solicitou o Setor O ativo na Figura 36(a), a próxima CTU será codificada com o Setor O ativo, conforme a Figura 36(b).

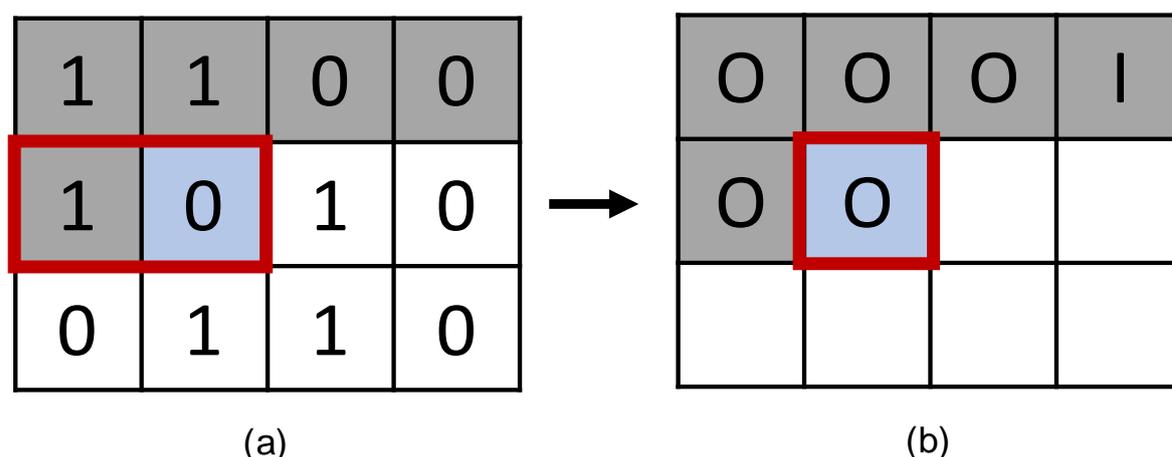


Figura 36. Ilustração da solução GD ESO: (a) matriz de referência e (b) setores ativados

5.7 Gerenciamento por meio do *Blur* (GD *Blur*)

O gerenciamento dinâmico baseado no *Blur*, verifica se pelo menos uma das CTUs vizinhas já codificada do quadro atual solicitou o Setor O ativo em sua codificação. Se sim, a CTU atual também recebe o Setor O habilitado para sua codificação. As solicitações das CTUs vizinhas são avaliadas conforme a matriz de referência ilustrada na Figura 37(a). Assim, somente as CTUs já codificadas do quadro atual são analisadas. No caso da Figura 37(a), como uma das CTUs vizinhas à próxima CTU a ser codificada solicitou o Setor O ativo, esta será codificada com o Setor O ativo, de acordo com a Figura 37(b).

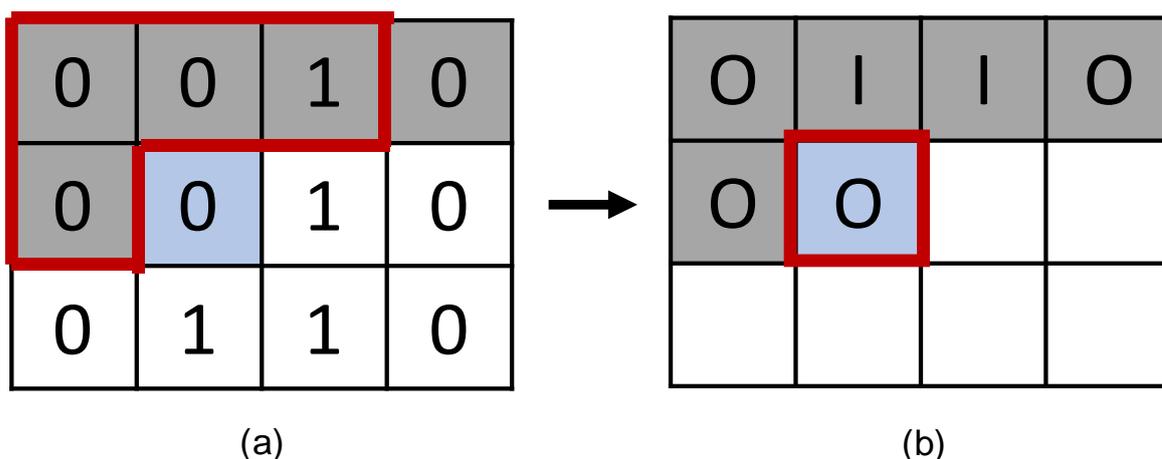


Figura 37. Ilustração da solução GD *Blur*: (a) matriz de referência e (b) setores ativados

5.8 Gerenciamento pela janela de CTUs (GD JCTU)

A estratégia que considera uma janela de CTUs funciona da seguinte forma: se uma ou mais das três CTUs anteriores solicitar o Setor O ativo, aquela CTU também recebe o Setor O ativo. Dessa forma, essa solução consiste em uma janela deslizante, onde as três CTUs imediatamente anteriores à atual vão continuar sendo avaliadas. Se nenhuma das três CTUs anteriores solicitar o Setor O ativo, é verificado se a CTU colocalizada no quadro anterior solicitou o Setor O ativo. Se sim, a CTU atual recebe o Setor O ativo. Na Figura 38, há uma ilustração dessa solução. Como uma das CTUs anteriores solicitou o Setor O na Figura 38(a), a CTU atual será codificada com o Setor O ativo, como ilustrado na Figura 39(b).

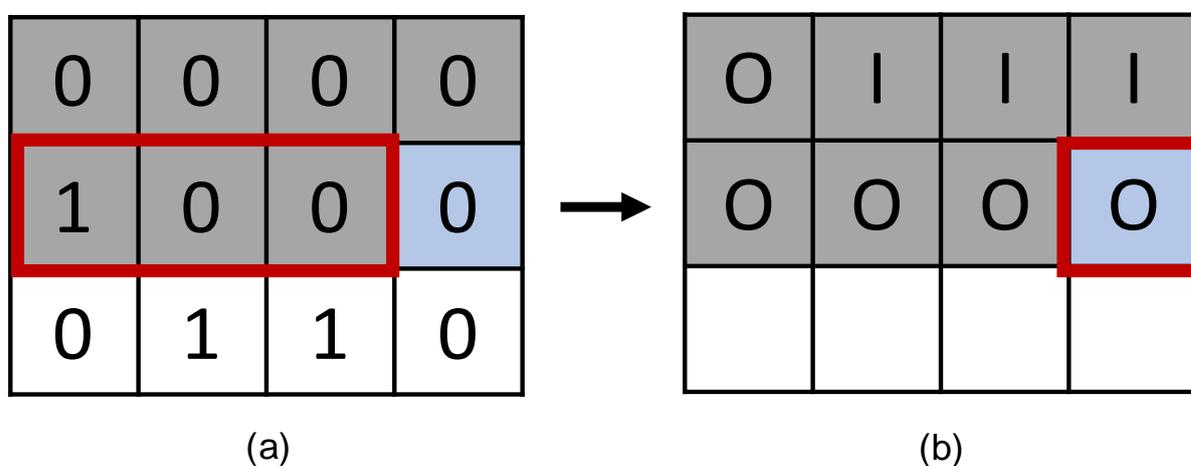


Figura 38. Ilustração da solução GD JCTU: (a) matriz de referência e (b) setores ativados

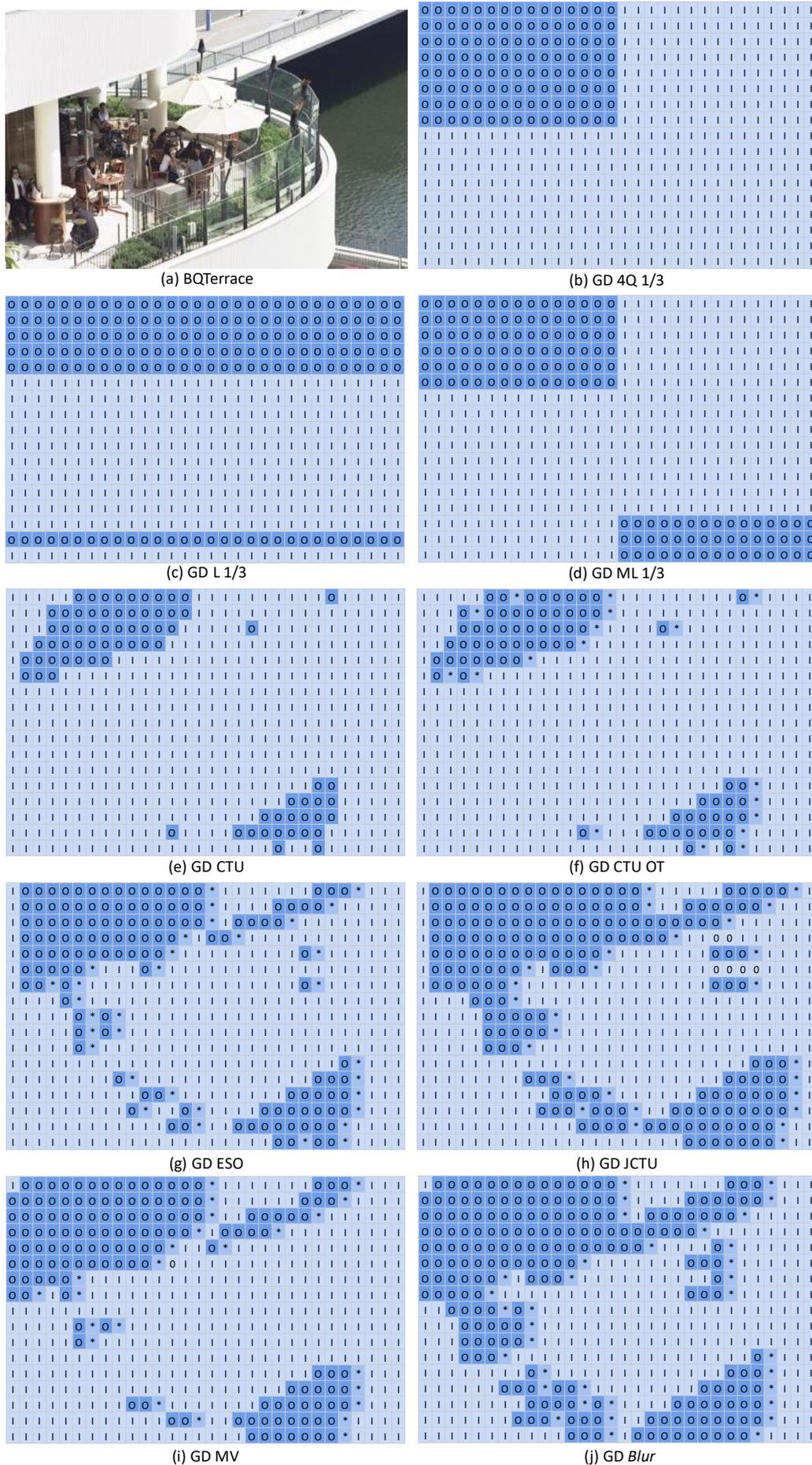


Figura 39. Gerenciamentos dinâmicos para codificar o Quadro 2 do vídeo BQTerrace

A Figura 39 resume, para a codificação do Quadro 2 do vídeo *BQTerrace*, as decisões tomadas pelo gerenciador dinâmico de estimação de movimento nas estratégias abordadas, incluindo as soluções que se baseiam em informações do quadro atual.

5.9 Modelo para avaliação de consumo de energia

O sistema adotado e avaliado neste trabalho envolveu o uso do compressor de quadros de referência sem perdas DDRFC, o reuso de dados *Level/C* e o gerenciador dinâmico da ME (vide Figura 19). Assim, o gerenciador proposto neste trabalho foi implementado em SystemC (SystemC, 2018), um conjunto de classes e macros para C++ com foco em modelagem a nível de sistema, e integrado com o modelo de avaliação de consumo de energia proposto em (AMARAL, 2015).

O modelo de avaliação de consumo de energia calcula o consumo de energia total de sistemas de codificação de vídeo que apresentam diferentes características, tais como níveis de reuso de dados, hierarquias de memória e compressores de quadros de referência. A entrada deste modelo é o *trace* do algoritmo TZS (vide Figura 9), com as informações detalhadas de todos os blocos requisitados durante a ME. A saída do modelo consiste no consumo de energia da solução avaliada.

No entanto, na primeira versão do modelo, apenas o consumo dinâmico dos módulos era calculado. Os cálculos necessários para a obtenção deste consumo seguem as equações apresentadas no capítulo 2 e consideram o caminho dos dados no sistema. Assim, o modelo retornava o consumo de energia proveniente dos custos de codificar os quadros pelo DDRFC, armazenar esses dados na memória externa, ler e decodificar esses dados, armazenar na memória interna e ler da memória interna, seguindo as requisições do TZS.

Como este trabalho propõe um gerenciamento dinâmico de estimação de movimento a partir de uma memória interna que permite *power gating* a nível de setores, o monitoramento do consumo estático dessa memória se torna relevante. Isso porque, ao desligar setores da memória, uma economia de energia no consumo estático da memória interna é obtida. A fim de quantificar essa economia, modificações foram feitas no modelo para permitir o cálculo do consumo estático sobre os ciclos necessários para codificar cada CTU.

Desta forma, foi considerada neste estudo uma arquitetura de ME que recebe 512 bytes como entrada e opera a uma frequência de 100 MHz (AFONSO, *et al.*, 2018). O número de ciclos necessários para processar um bloco segue a Equação 7, onde TB se refere ao tamanho do bloco, $nLinhas$, ao número de linhas do bloco e $nCand$ ao número de blocos comparados pelo TZS.

$$Ciclos = 1 + \log_2 TB + nLinhas + \log_2 nCand \quad (7)$$

Com isso, a memória interna foi projetada para receber e entregar 64 bytes por ciclo e operar a uma frequência de 800 MHz, compondo a entrada da arquitetura de ME adotada. Esse tamanho de entrada também condiz com a granularidade do compressor DDRFC, que neste trabalho comprime blocos de tamanho 8x8, e com o menor tamanho de bloco utilizado na ME do HEVC, também 8x8.

Por fim, o cálculo do consumo estático considera o número de ciclos necessários para processar cada CTU e também o tempo de processamento em que o setor requisitado ficou ativo na memória interna. A Equação 8 reflete o cálculo realizado para a obtenção do consumo estático total da memória interna, realizado a partir do somatório do consumo estático de cada setor. Nesta equação, CT se refere ao tempo do ciclo da arquitetura da ME, SEC , ao consumo estático referente ao número de células SRAM ativas na memória e n , ao número máximo de setores que podem ser ativados na memória.

$$ConsumoEstático = \sum_{i=0}^{i < n} (CT \times Ciclos_i \times SEC_i) \quad (8)$$

Para gerar os resultados de consumo, uma *Low Power Double Data Rate Synchronous Dynamic Random Access Memory* (LPDDR SDRAM) da Micron (MICRON, 2017) foi considerada. Esta memória consome, por *byte*, 119,7 pJ e 116pJ para operações de leitura e escrita, respectivamente. A ferramenta Cacti 6.5 da HP (LABS, 2017) foi usada para simular SPMs de tamanho equivalente a área de busca sem cortes e aos cortes fixos realizados. As informações destas memórias se encontram na Tabela 8. Quando as soluções de ajuste dinâmico são consideradas, os consumos de energia quando o Setor 0 está ligado equivalem ao do CF 33,35%. No entanto, quando este setor é desligado e apenas o Setor 1 permanece ativo, como o Setor 1 representa apenas 26,82% do tamanho total do CF 33,35%, a energia estática consumida é reduzida para 0,71 mJ/s, de acordo com a alteração no número de células SRAM ativas na memória. O consumo dinâmico não é alterado, dado que

Tabela 8. Informações gerais das SPMs

	SR [-64, +63]	CF 33,35%	CF 65,60%	CF 81,36%
Número de bancos	576	384	200	103
Access time (ns)	0,75	0,67	0,68	0,62
Cycle time (ns)	1,21	1,22	1,23	1,18
Consumo dinâmico (pJ/byte)	0,09	0,08	0,05	0,03
Consumo estático (mJ/s)	4,56	2,64	1,38	0,69

este se refere ao chaveamento das células acessadas e seus barramentos e amplificadores. Por fim, o DDRFC consome 4,14 pJ e 6,77 pJ, por *byte*, na compressão e descompressão, respectivamente.

5.10 Resultados

As soluções de gerenciamento dinâmico da ME foram avaliadas em eficiência de codificação e em redução de consumo de energia, medida por meio do modelo de avaliação de consumo de energia. Deste modo, ainda que o gerenciador dinâmico esteja integrado com uma solução de reuso de dados e um compressor de quadros de referência, somente os ganhos referentes ao gerenciador e às soluções de gerenciamento dinâmico serão apresentados e discutidos neste capítulo. Os resultados de BD-Rate estão apresentados no gráfico da Figura 40. Para a geração destes resultados foram avaliados os vídeos *BasketballDrive*, *BQTerrace*, *Cactus*, *Kimono*, *ParkScene* e *Tennis*. Neste gráfico, o corte fixo mais agressivo, CF 81,36% foi removido, já que seus resultados foram muito discrepantes em relação às demais soluções. Além disso, por alcançaram as maiores variações e média em eficiência de codificação, este resultado não é interessante no escopo deste trabalho que busca reduzir o consumo de energia, mas atingir baixo impacto em eficiência de codificação. Resultados detalhados de BD-Rate se encontram no Apêndice A.

Por meio dos resultados de BD-Rate, nota-se que cortes fixos mais agressivos, como o CF 66,65% e o CF 81,36% se destacaram como as soluções que tiveram os maiores impactos e variação em eficiência de codificação. Os cortes fixos condicionam que um vídeo inteiro seja codificado com uma mesma área de busca. Porém, quanto mais agressivo este corte, maior a dificuldade do encontro do melhor

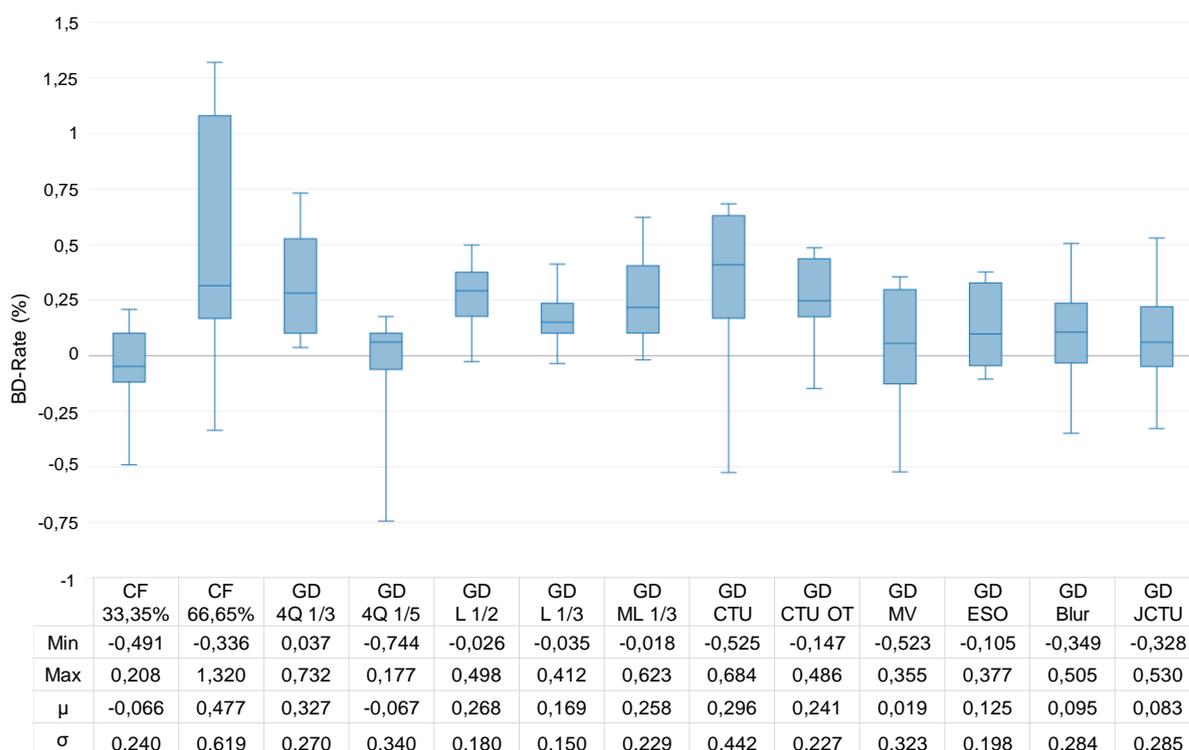


Figura 40. Resultados de BD-Rate

casamento pelo algoritmo da ME. Por outro lado, as estratégias de gerenciamento dinâmico se mantiveram muito próximas dos resultados de BD-Rate obtidos pelo CF 33,35%. Isso se deve ao fato de que estas soluções são capazes de explorar a necessidade de codificação de diferentes regiões de um mesmo vídeo.

Granularidades maiores criam uma maior dependência entre regiões de um quadro e acarretam em um maior número de erros, que ocorrem quando uma CTU solicita o Setor O ligado, mas recebe apenas o Setor I ou o contrário. Em termos de eficiência de codificação, a maior variação ocorre quando a CTU solicita o Setor O ativo para sua codificação, mas o gerenciamento dinâmico disponibiliza apenas o Setor I, com menos amostras disponíveis para comparação. Por outro lado, em termos de eficiência energética, é sempre melhor manter o menor número de setores ativos na memória interna, atingindo uma redução no consumo estático e no número de amostras acessadas e, assim, reduzindo também o consumo dinâmico.

Pelos resultados médios de BD-Rate, no entanto, é possível perceber que os melhores resultados são obtidos quando granularidades menores com informações baseadas na codificação das CTUs vizinhas do mesmo quadro são utilizadas, como no caso do GD MV, GD ESO, GD *Blur* e GD JCTU. Essas soluções conseguem atingir

uma menor taxa de erros, pois contam com as informações das CTUs codificadas do quadro passado e do quadro atual.

Dentre as soluções de gerenciamento testadas, o vídeo *ParkScene* foi o que apresentou a maior variação de BD-Rate. Isso ocorre porque esse vídeo apresenta baixa movimentação e, portanto, o Setor O fica desligado um maior número de vezes. As escolhas do TZS para o melhor casamento nos casos das soluções de gerenciamento dinâmico por quadrantes e por CTUs foram positivas, nestes casos o vídeo *ParkScene* foi o que apresentou os menores valores de BD-Rate. No entanto, no caso das soluções GD L 1/3 e GD ML 1/3, esse mesmo vídeo foi o que atingiu as maiores taxas de BD-Rate.

O vídeo *Tennis*, por outro lado, é um vídeo de alta movimentação que apresentou as menores variações de BD-Rate. Em algumas soluções de gerenciamento dinâmico com granularidades maiores, como o GD 4Q 1/5, GD 4Q 1/3, GD L 1/3, o Setor O nunca chegou a ser desligado. Neste caso, o vídeo apresentou os mesmos resultados de BD-Rate do CF 33,35%. Nas outras soluções de gerenciamento dinâmico para este mesmo vídeo, com o desligamento do Setor O, as escolhas dos melhores casamentos pelo TZS acarretaram em variações maiores no BD-Rate, ainda que pequenas.

A relação entre os consumos de energia estático e dinâmico se encontram no gráfico da Figura 41. Em memórias o consumo estático tende a ser predominante. No entanto, devido ao grande número de comparações realizados na etapa da ME pelo

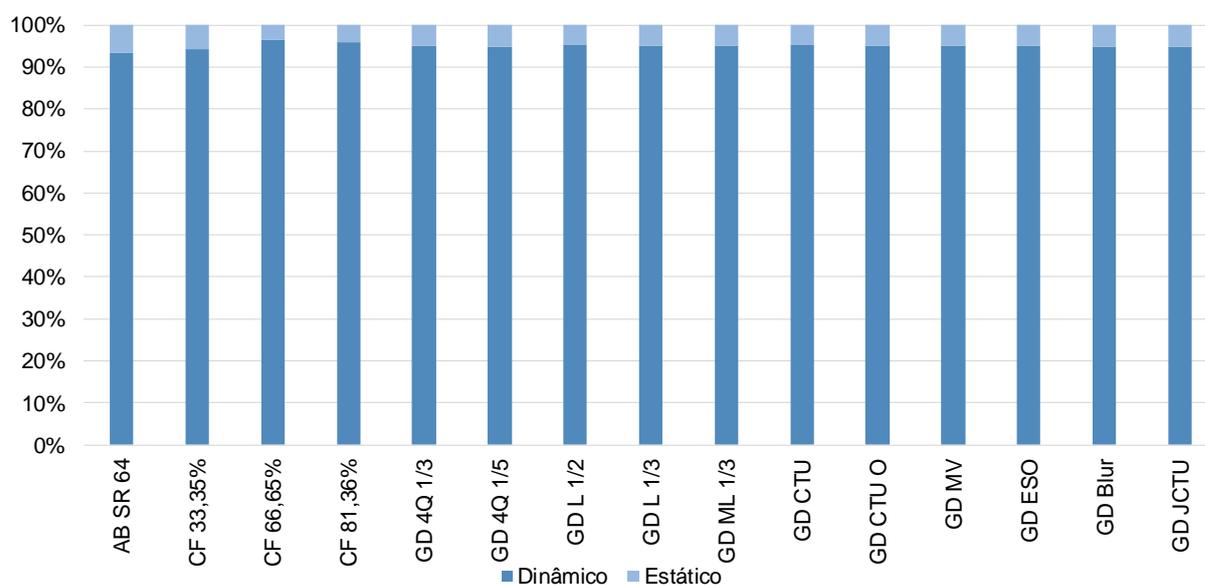


Figura 41. Resultados entre os consumos estático e dinâmico da SPM

TZS, os resultados de redução de consumo total são altamente influenciados pelos resultados de consumo dinâmico. Assim, por meio das relações apresentadas por todas as soluções avaliadas no gráfico da Figura 41, nota-se que o consumo estático representa menos de 10% do consumo total, que é dominado pelo consumo dinâmico.

Desta forma, os resultados de reduções de consumo de energia (dinâmico, estático e total) alcançadas pelas soluções de gerenciamento dinâmico e cortes fixos se encontram no gráfico da **Erro! Fonte de referência não encontrada.** Para resultados detalhados de consumo e redução de consumo, consultar o Apêndice B. A solução que apresenta a menor redução de consumo de energia é o CF 33,35%, pois apresenta, entre os cortes fixos, a maior região ligada na memória e permite um maior número de comparações. Desse modo, o CF 81,36% é o que consegue atingir as maiores taxas de redução de consumo dinâmico, por ser o corte mais agressivo e deixar o menor número de células SRAM ativas na SPM. As reduções de consumo dinâmico alcançadas pelas estratégias de gerenciamento dinâmico se mantiveram entre as atingidas pelos CF 33,35% e CF 81,36%, conforme esperado. Além disso, as soluções de gerenciamento dinâmico também conseguiram, em média, reduções maiores em consumo dinâmico do que o CF 65,60%. Em relação ao gerenciamento dinâmico, o CF 65,60% obteve as maiores reduções de consumo dinâmico nos vídeos de alta movimentação, *Basketball*, *Cactus* e *Tennis*, onde o Setor O é requisitado um maior número de vezes. No entanto, em vídeos de baixa movimentação, como as

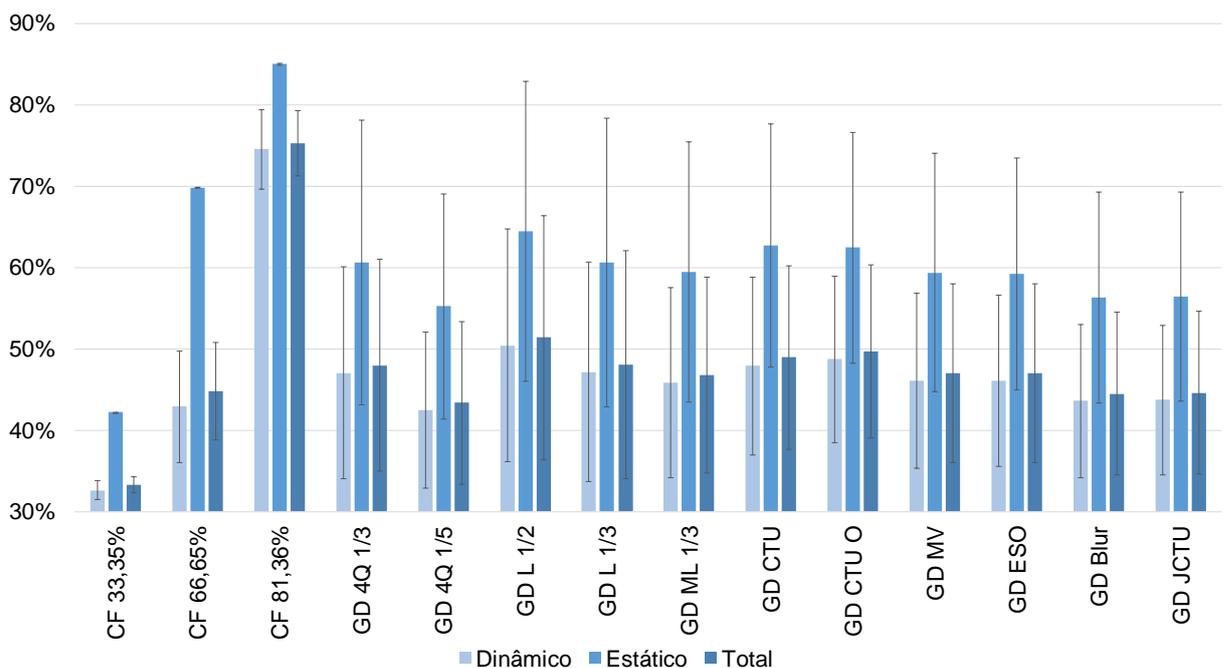


Figura 42. Resultados de redução de consumo das soluções de GD

sequências *BQTerrace*, *Kimono* e *ParkScene*, as soluções de gerenciamento dinâmico atingiram reduções maiores, pois o Setor O pode se manter desligado um maior número de vezes nestes casos.

Os resultados de consumo estático apresentam uma mesma relação. O corte mais agressivo é o que atinge as maiores reduções de consumo estático e as soluções de gerenciamento dinâmico se mantêm entre as taxas de redução atingidas pelos CF 33,35% e CF 81,36%. No caso do consumo estático, a redução alcançada pelas soluções de gerenciamento dinâmico também é maior do que a atingida pelo CF 66,65% em vídeos de baixa movimentação. Isso leva as estratégias de gerenciamento dinâmico a atingir reduções maiores do que o CF 66,65% na média.

Dentre as soluções de gerenciamento dinâmico, o GD L 1/2 foi o que atingiu as maiores reduções em consumo de energia, chegando a uma redução média de consumo de energia de 51,36%. Isso se deve ao fato de que esta estratégia de GD é a menos permissiva entre as abordadas. Para manter o Setor O ativo para toda uma linha em um quadro a ser codificado, é necessário que um número igual ou maior do que a metade de CTUs já codificadas na linha colocalizada no quadro de referência tenham requisitado o Setor O ativo. Essa condição faz com que o GD L 1/2 atinja uma das maiores variações de BD-Rate entre as soluções de gerenciamento dinâmico.

Por outro lado, a solução mais permissiva, o GD 4Q 1/5, que divide o quadro em quatro quadrantes e determina que o Setor O estará ativo para codificar um determinado quadrante se um quinto ou mais das CTUs já codificadas do quadrante colocalizado no quadro anterior solicitar o Setor O, é a que atinge as menores reduções em consumo de energia, em média 43,36%. Esse fato contribui para atingir a variação média em BD-Rate mais próxima da obtida pelo CF 33,35%.

Por fim, as soluções de gerenciamento dinâmico são muito vantajosas quando os eixos eficiência de codificação e redução de consumo de energia são considerados. Ao permitir que CTUs que necessitam de um menor esforço computacional na ME tenham o Setor O desligado na memória, uma redução energética significativa pode ser alcançada. Ao mesmo tempo, ao ativar o Setor O na memória para a codificação de CTUs que exigem um maior esforço computacional, as soluções de gerenciamento dinâmico evitam grandes degradações em eficiência de codificação.

Todos os resultados apresentados até aqui consideram o mesmo conjunto de sequencias vídeos. Este conjunto foi usado para gerar a análise dos acessos do TZS

dentro da área de busca, nas tomadas de decisões do projeto da memória de busca e na geração dos resultados das soluções avaliadas. Desta forma, é necessário validar essas soluções com um conjunto de sequencias de vídeos diferentes, de resoluções HD 1080p e superiores. Os resultados finais se encontram no capítulo 6 desta dissertação e consideraram apenas as soluções desenvolvidas que obtiveram as relações mais vantajosas entre eficiência de codificação e redução de consumo de energia.

6 RESULTADOS FINAIS E DISCUSSÕES

Este capítulo visa expandir os resultados obtidos no capítulo anterior e validar algumas soluções com um novo conjunto de sequências de vídeo de resolução HD 1080p e também para vídeos de resolução *Ultra High Definition* (UHD) 2160p (4096x2160 ou 3840x2160 pixels). O conjunto de vídeos com resolução UHD 2160p adotado neste capítulo compreende os vídeos das classes A1 e A2 das Condições Comuns de Teste (CCTs) (SHARMAN e KARSTEN, 2017). Esses vídeos são: *Campfire*, *Drums*, *Tango*, *ToddlerFountain*, *CatRobot*, *DaylightRoad*, *RollerCoaster* e *TrafficFlow*. As sequências 1080p escolhidas foram *BlueSky*, *InToTree*, *PedestrianArea*, *RushHour*, *Sunflower* e *Tractor*.

Para a escolha das soluções avaliadas neste capítulo, os resultados médios de BD-Rate e de redução energética de cada solução foram verificados. Esta relação entre BD-Rate e redução energética total, para todas as soluções, é encontrada no gráfico da Figura 43. Por meio do gráfico obtido, uma frente de Pareto foi gerada a fim de estabelecer quais são as soluções mais promissoras, ou seja, as que apresentaram as melhores relações entre os dois eixos. As soluções retornadas pela frente de Pareto e indicadas com um losango na Figura 43 foram: GD 4Q 1/5, GD MV, GD L 1/3, GD CTU OT GD L 1/2 e CF 81,36%.

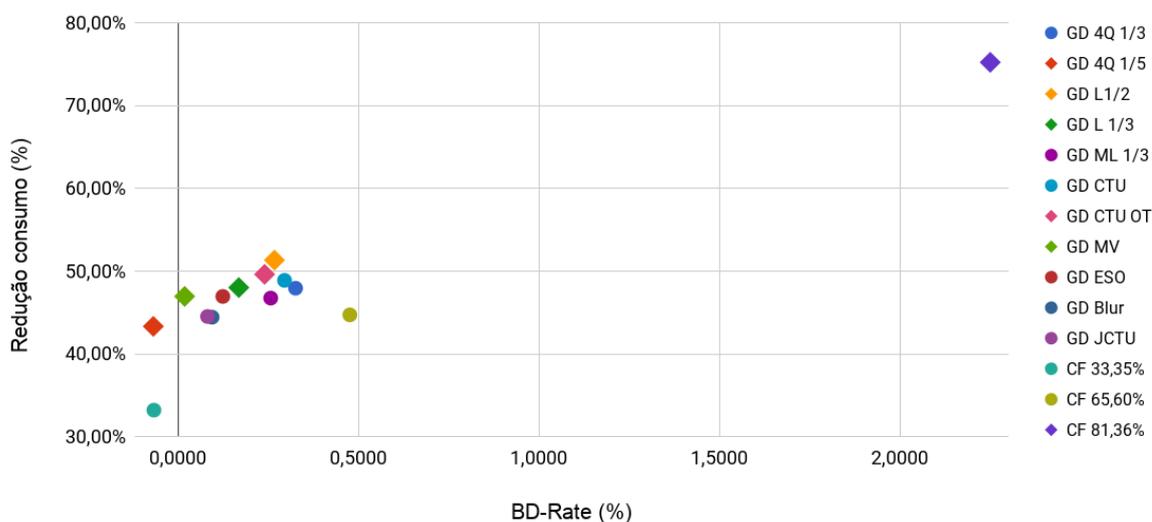


Figura 43. Relação entre os resultados de redução do consumo e BD-Rate

Para facilitar a análise das cinco soluções retornadas pela frente de Pareto, essas foram listadas na Tabela 9, junto com o resultado médio de BD-Rate e a taxa média de redução de consumo de energia total.

Tabela 9. Resultados de BD-Rate e redução de consumo das soluções retornadas pela frente de Pareto

Solução	BD-Rate (%)	Redução de Consumo (%)
CF 81,36%	2,2510	75,24%
GD L 1/2	0,2681	51,36%
GD CTU OT	0,2407	49,64%
GD L 1/3	0,1691	48,04%
GD MV	0,0193	46,98%
GD 4Q 1/5	-0,0674	43,36%

Os resultados na Tabela 9 foram organizados de forma decrescente. Por meio desta tabela, nota-se que os resultados de BD-Rate variam entre -0,0674% e 2,2510% e que os resultados de redução de consumo de energia variam entre 43,36% e 75,24%. No entanto, a solução GD L 1/3 traz um aumento considerável em BD-Rate quando comparado com a solução GD MV e um aumento negligível em redução de consumo de energia (pouco mais de 1%). Além disso, a solução GD CTU OT provoca uma redução em consumo de energia pouco menor do que a solução GD L 1/2, mas o BD-Rate médio não apresenta uma diferença vantajosa. Desse modo, as soluções GD L 1/3 e GD CTU OT não foram exploradas no restante deste capítulo.

Os resultados finais de BD-Rate para as soluções avaliadas constam na Tabela 10. Estes resultados são sempre comparados com a área de busca original sem cortes (SR [-64, +63]). Por meio desta tabela, nota-se que enquanto o CF 33,35% continuou apresentando pouca variação nos resultados de eficiência de codificação em relação à área de busca original e que o maior impacto em BD-Rate ocorreu quando o CF 81,36% foi adotado. Com exceção dos vídeos *Drums* e *CatRobot*, de baixa movimentação, todos os outros vídeos apresentaram variações em BD-Rate maiores do que 2% quando o corte fixo mais agressivo foi utilizado. No caso de vídeos de resoluções maiores, caso das sequências *Tango*, *DaylightRoad*, *RollerCoaster* e *TrafficFlow*, isso é esperado. Isso porque quadros de resoluções maiores conseguem representar um maior nível de detalhes em suas imagens e isso faz com que o encontro do melhor casamento seja dificultado, pois o algoritmo de estimação de

Tabela 10. Resultados de BD-Rate expandidos

Sequência	CF 33,35%	CF 81,36%	GD 4Q 1/5	GD MV	GD L 1/2
CampfireParty	0,9721	5,1804	1,3480	1,3351	4,5111
Drums	0,4485	1,7995	0,5963	0,6539	1,0228
Tango	0,5615	10,4930	0,3535	0,7405	0,9278
ToddlerFountain	0,6135	2,6658	0,6135	0,5073	0,6512
CatRobot	0,0921	0,9476	0,2510	0,0481	0,7738
DaylightRoad	0,6565	13,9640	0,6781	0,7592	4,5446
RollerCoaster	-0,5191	27,0270	-0,5191	-0,3753	2,4320
TrafficFlow	-0,3684	11,0740	2,6345	1,8329	10,3260
BlueSky	-0,4166	2,7317	-0,1333	0,4882	-0,2352
InToTree	2,3818	15,0160	2,3818	-0,0016	3,1952
PedestrianArea	0,5476	8,2704	0,2067	0,3204	2,6335
RushHour	-0,0046	2,6900	0,6374	0,4411	1,0625
Sunflower	-0,0280	4,8132	0,4156	0,1805	-0,1046
Tractor	0,5030	3,2751	0,5030	-0,0043	0,6583
Média	0,3886	7,8534	0,7119	0,4947	2,3142
Mínimo	-0,5191	0,9476	-0,5191	-0,3753	-0,2352
Máximo	2,3818	27,0270	2,6345	1,8329	10,3260
Desvio Padrão	0,7322	7,2072	0,8716	0,5709	2,7772

movimento necessita realizar um maior número de comparações. Como o corte mais agressivo lida com uma área de busca mais limitada, o TZS não consegue fazer as comparações ideais para esses vídeos. Essa ocorrência também é comum em vídeos de alta textura, como a sequência *InToTree* de resolução 1080p, que também alcançou uma variação alta em BD-Rate com o CF 81,36%.

Assim, as estratégias de gerenciamento dinâmico alcançaram taxas de BD-Rate mais vantajosas do que o CF 81,36% para todos os vídeos. No entanto, o GD 4Q 1/5, por ser mais permissivo, não chegou a desligar o Setor O para os vídeos *InToTree* e *Tractor*, de alta textura e movimentação, e com isso chegou aos mesmos valores de BD-Rate que o CF 33,35%. Como esperado, no entanto, a maior parte dos resultados de BD-Rate se mantiveram entre os resultados do CF 33,35% e CF

81,36%. As exceções são destacadas em negrito na Tabela 10. Estes valores que ficaram fora do intervalo apresentado pelos cortes fixos se devem às diferenças da escolha do melhor casamento pelo TZS, forçadas pelo desligamento e ligamento do Setor O e pelo fato de que o algoritmo sempre considera apenas os valores de similaridade entre blocos. Com isso, outros valores que impactam a codificação como o tamanho dos vetores de movimento e as características dos vetores de movimento dos blocos vizinhos não são observados. Desta forma, blocos com similaridades muito próximas, mas tamanhos de vetor de movimento menores podem gerar um *bitrate* menor.

Soluções de gerenciamento dinâmico com granularidades pequenas, caso do GD MV, por permitirem uma independência maior de ligamento/desligamento do Setor O entre as CTUs apresentam uma maior ocorrência de valores de BD-Rate fora do intervalo apresentado pelos cortes fixos. Além disso, como esta solução se baseia em informações de CTUs vizinhas, já codificadas no quadro atual e passado, para decidir a situação do Setor O para a CTU sendo codificada, essa solução consegue alcançar variações baixas em BD-Rate. No caso dos vídeos testados neste capítulo, a GD MV foi a solução de gerenciamento de ME com a taxa média de BD-Rate mais próxima a do CF 33,35% e também a solução que apresentou o menor desvio padrão.

Por outro lado, a solução GD L 1/2, por ter um nível de exigência maior na decisão de ligar o Setor O do que as demais, é a estratégia de gerenciamento dinâmico que atinge a maior variação de BD-Rate. Como esta solução mantém apenas o Setor I ativo na memória interna um maior número de vezes, alguns resultados de BD-Rate se aproximaram dos resultados obtidos pelo CF 81,36%. Isso é notado especialmente nos casos dos vídeos 2160p *CampfireParty*, *Drums*, *CatRobot* e *TrafficFlow*, que exigem um menor esforço computacional da ME (menor número de comparações) do que outros vídeos.

Os resultados finais de redução de consumo de energia se encontram no gráfico da Figura 44. Para resultados detalhados, consulte o Apêndice C. Com os resultados obtidos, é possível perceber que a relação entre os resultados de consumo dinâmico obtidos no capítulo anterior se manteve para o conjunto de vídeos utilizado neste capítulo. Desse modo, os cortes fixos foram as soluções que apresentaram a menor variação entre seus resultados. Isso ocorre porque estes cortes não exploram o esforço computacional demandado pelas CTUs na codificação dos vídeos. Assim, o CF 33,35%, por apresentar a maior área de busca disponível na memória interna e,

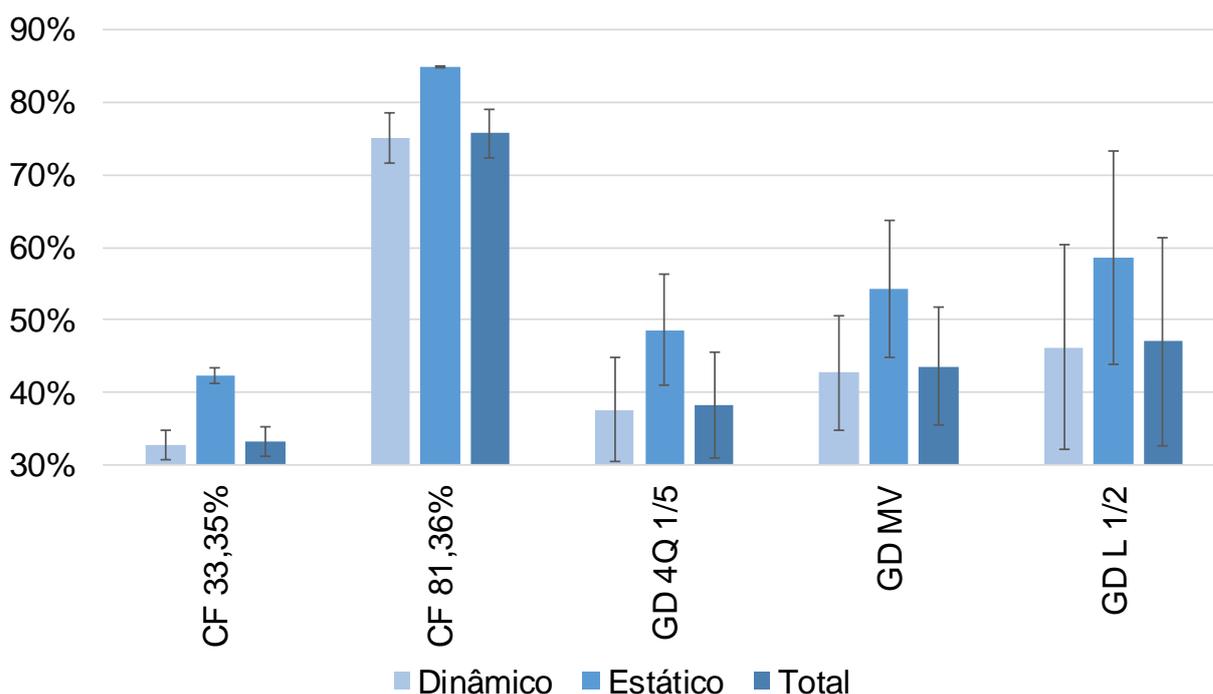


Figura 44. Resultados finais de redução de consumo das soluções de GD

assim, permitir um maior número de comparações entre blocos durante a ME, é o que apresenta a menor redução energética em consumo dinâmico (32,59%). Da mesma forma, o CF 81,36%, por apresentar a menor área de busca disponível na memória interna, é o que apresentar os menores valores de consumo dinâmico. Em média, o CF 81,36% reduz 74,51% do consumo dinâmico quando comparado com a área de busca original sem cortes. No entanto, enquanto os valores de BD-Rate do CF 33,35% apresentam pouco impacto em eficiência de codificação, os resultados alcançados pelo CF 81,36% apresentam variações mais agressivas.

Por outro lado, os resultados médios de BD-Rate das estratégias de gerenciamento dinâmico da estimação de movimento ficam próximos das alcançadas pelo CF 33,35% e possuem a vantagem de alcançar taxas de redução energética maiores. A solução GD 4Q 1/5, por permitir que o Setor O fique ligado um maior número de vezes, foi a que obteve a menor redução de consumo dinâmico quando comparada com as demais soluções de gerenciamento dinâmico. No caso dos vídeos *InToTree* e *Tractor*, o Setor O nunca foi desligado e, com isso, os resultados de consumo dinâmico foram iguais aos alcançados pelo CF 33,35%.

O GD MV, por sua vez, teve resultados de redução de consumo de energia maiores do que os alcançados pela solução GD 4Q 1/5 e menores do que os alcançados pela solução GD L 1/2. Isso é explicado pelo fato de que o GD 4Q 1/5

tende a deixar o Setor O ligado na memória para um maior número de CTUs entre as soluções de gerenciamento dinâmico, enquanto o GD L 1/2 tende a deixar o Setor O desligado na memória interna um maior número de vezes. Desta forma, o GD L 1/2 foi a estratégia de gerenciamento dinâmico que atingiu as maiores taxas de redução de consumo dinâmico. Esse ganho foi notado principalmente nos vídeos *CampfireParty*, *Drums*, *CatRobot* e *TrafficFlow*, onde o Setor O ficou desligado um maior número de vezes. No entanto, como discutido anteriormente, estes vídeos foram os que obtiveram as maiores variações em BD-Rate com a solução GD L 1/2.

Como no capítulo anterior, nota-se que os resultados de consumo estático alcançam reduções muito maiores do que os resultados de consumo dinâmico. No entanto, o TZS faz inúmeros acessos à memória durante a codificação de um vídeo. E esse custo volumoso de acessos acaba superando o custo de energia de deixar a memória ligada durante o processamento do vídeo. Essa relação entre os consumos é apresentada na Figura 45. Novamente de acordo com o capítulo anterior, os resultados de consumo estático representam menos de 10% do consumo total.

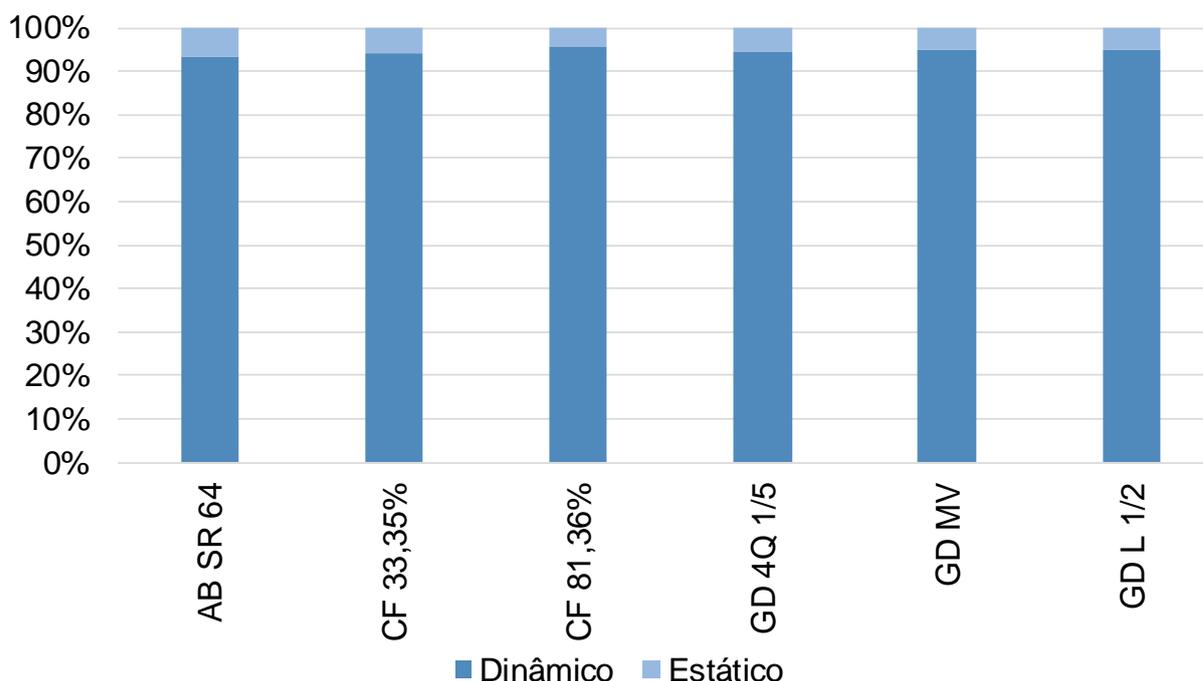


Figura 45. Relações entre os consumos estático e dinâmico da SPM dos resultados finais

Conforme os resultados de redução de consumo dinâmico, a redução de consumo estático apresenta o mesmo comportamento. O CF 33,35% é o que apresenta a menor redução, enquanto o CF 81,36% atinge as maiores reduções.

Entre as soluções de gerenciamento dinâmico, a menor redução foi alcançada pelo GD 4Q 1/5 e a maior redução, pelo GD L 1/2.

Por fim, os resultados de redução do consumo de energia total aliados com os resultados de BD-Rate mostram que os vídeos se beneficiam de estratégias de gerenciamento dinâmico de estimação de movimento. Pois, ao explorar as características de regiões do vídeo durante a codificação do mesmo, é possível desligar setores da memória quando estes se tornam pouco relevantes para a ME. Deste modo, ao desligar setores na memória é possível obter uma boa relação entre consumo de energia e eficiência de codificação.

Da mesma forma, reduções de consumo mais altas tendem a causar impactos maiores em eficiência de codificação. Assim, o GD L 1/2 conseguiu atingir uma redução média em consumo de energia de 47,04% quando comparado com a área de busca original e obteve uma variação em BD-Rate média de 2,31%. O GD MV, por outro lado, apresentou uma redução de consumo de 43,59% e uma variação média em BD-Rate de 0,49% para os vídeos testados.

A relação entre os resultados de BD-Rate e os resultados de redução de energia se encontram na Figura 46. No gráfico desta figura, os resultados de BD-Rate em laranja, que obedecem a escala da direita, estão sobrepostos sobre os resultados de redução de consumo de energia em azul, que obedecem a escala da esquerda.

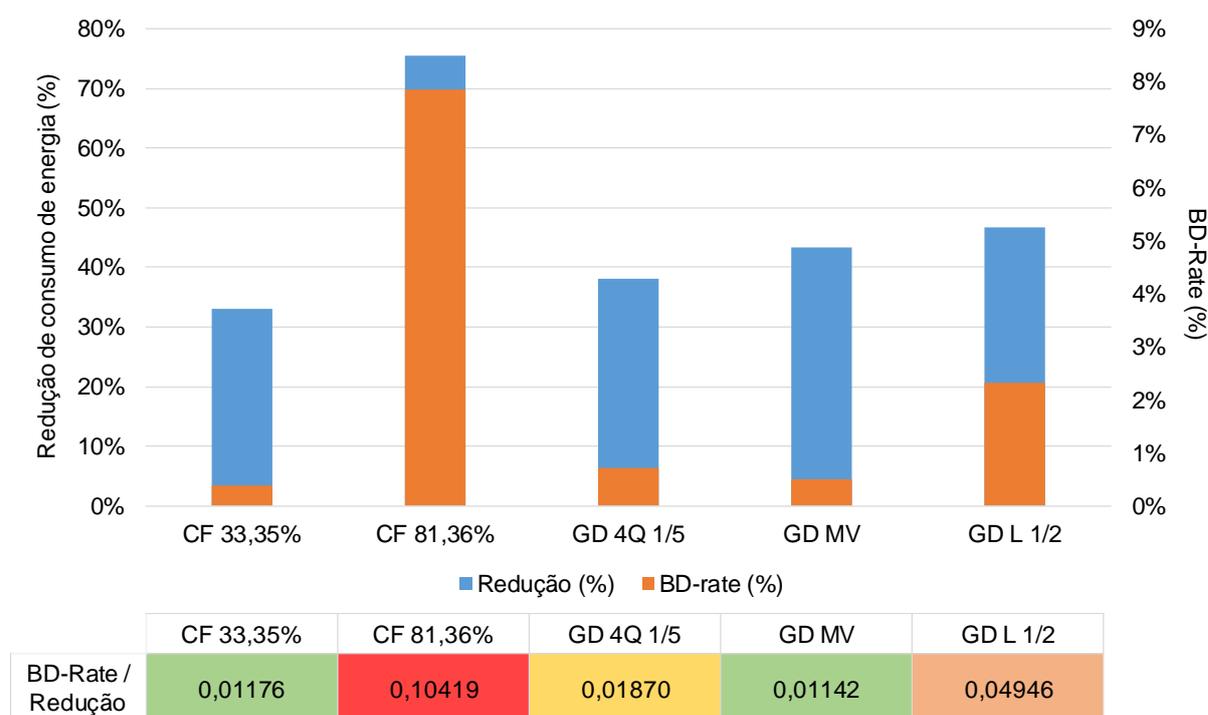


Figura 46. Relações entre os resultados finais de redução de consumo e BD-Rate

Desta forma, as soluções que apresentam a melhor relação entre estes dois eixos são as que contam com a maior barra azul (redução de energia) e a menor barra laranja (impacto em BD-Rate). Com isto, três soluções se destacam. São elas o CF 33,35%, o GD 4Q 1/5 e o GD MV. Dentre as três, o GD MV é a que apresenta a maior redução de energia e o menor impacto em BD-Rate, após o CF 33,35%. Abaixo do gráfico na Figura 46, consta a razão entre os resultados de BD-Rate e a redução de consumo de energia para cada uma das soluções avaliadas. Os resultados desta relação obedecem a um mapa de calor, onde os impactos mais baixos recebem cor verde e os mais altos, cor vermelha. É importante observar que o consumo de energia do gerenciador de memória, que liga e desliga o setor mais externo, não foi contabilizado neste estudo para nenhuma das soluções avaliadas.

Ainda, tendo em vista os resultados alcançados por cada vídeo, uma estratégia ainda melhor poderia ser obtida por meio da união de todas estas técnicas de cortes fixos e gerenciamento dinâmico estudadas até então. Assim, um novo controlador poderia avaliar, durante a codificação de um vídeo, a relação entre os valores de *bitrate* e PSNR e decidir, dinamicamente, qual estratégia de gerenciamento dinâmico ou corte fixo adotar para um determinado conjunto de quadros ou CTUs. Essa decisão se basearia em objetivos como menor impacto em qualidade ou maior redução energética e poderia explorar melhor o fato de que diferentes vídeos se adaptaram melhor com diferentes soluções. Os vídeos *ToddlerFountain*, *CatRobot*, *TrafficFlow*, *InToTree*, *RushHour*, *Sunflower* e *Tractor* obtiveram menores impactos em BD-Rate e maiores reduções de consumo quando codificados com o GD MV do que quando codificados com o GD 4Q 1/5, por exemplo.

6.1 Redução de consumo do sistema avaliado

Até o momento, foram discutidos os ganhos obtidos pelo uso dos cortes fixos e do gerenciamento dinâmico da estimação de movimento sem considerar os ganhos do reuso de dados *Level C* e do compressor de quadros de referência DDRFC, que atinge uma taxa média de compressão de 57,5%. No entanto, esses dois módulos continuaram atuando durante todo o estudo deste trabalho e, em conjunto com as estratégias de gerenciamento dinâmico e os cortes fixos, geraram os custos de energias discutidos até então. Com isso, a fim de avaliar o impacto do uso das três técnicas: reuso de dados, compressão de quadros de referência e gerenciamento

dinâmico da ME ou corte fixo, os resultados de consumo de energia das soluções exploradas neste capítulo foram comparados com a área de busca sem cortes quando o reuso de dados e a compressão de quadros de referência não são aplicados (SR [-64, +63]*). As reduções de consumo de energia quando as três técnicas são consideradas se encontram na Tabela 11. Nesta tabela também são inseridos os vídeos discutidos nos capítulos anteriores *BasketballDrive*, *BQTerrace*, *Cactus*, *Kimono*, *ParkScene* e *Tennis*.

No escopo do trabalho apresentado em (AMARAL, 2015), uma redução de consumo de energia de 97,8% foi alcançada com o uso das técnicas de reuso de dados e compressão de quadros de referência. No entanto, esse valor não é comparável com o alcançado neste trabalho, já que em (AMARAL, 2015) as técnicas utilizadas não causam impacto nos resultados de codificação. Neste trabalho, ao contrário, o impacto da eficiência de codificação é um dos principais eixos avaliados. Com isso, o módulo da estimação de movimento foi isolado. Assim, técnicas como *early skip* e a ocorrência de blocos intra em quadros inter foram desabilitados. Além disso, de forma a reduzir o número de acessos à memória, apenas um quadro de referência foi adotado neste trabalho e a bipredição também foi desabilitada. Por fim, embora este trabalho contabilize o consumo estático da memória interna, foram adotadas memórias que operam com frequências próximas a 800 MHz. Essa frequência de operação contribuiu para custos de energias de escrita e leitura menores do que quando comparados à memória adotada em (AMARAL, 2015). No trabalho (AMARAL, 2015), os custos de energias de leitura e escrita por *byte* na SPM foram 2,27pJ e 1,58pJ. Neste trabalho, estes custos equivalem a 0,09pJ em ambas as operações para a maior configuração de memória interna.

Com isso, os resultados da Tabela 11 mostram que o emprego de todas as técnicas em conjunto atinge reduções de consumo de energia muito elevadas quando comparada com a não adoção dessas técnicas. As altas reduções de consumo de energia se devem, principalmente, ao emprego do reuso de dados *Level C*, visto que uma enorme quantidade de leituras na memória deve ser feita durante a codificação de vídeo. Quando o reuso de dados é aplicado, um mesmo dado em uma área de busca não precisa ser acessado na memória externa sempre que requisitado pela ME. Aliado a este fato, estão os baixos gastos de energias nas operações de leitura e escrita da SPM quando comparado com a LPDDR SDRAM, tornando o *Level C* uma

Tabela 11. Resultado de consumo de energia total da SR [-64, +63]* e reduções de consumo de energia (%) obtidos pelo uso do *Level C*, DDRFC e CF ou GD

Sequência	SR [-64,+63]*	SR [-64,+63]	CF 33,35%	CF 81,36%	GD 4Q 1/5	GD MV	GD L 1/2
CampfireParty	87525,68	99,71%	99,81%	99,94%	99,85%	99,85%	99,90%
Drums	93601,28	99,70%	99,80%	99,93%	99,82%	99,83%	99,85%
Tango	151975,10	99,79%	99,86%	99,96%	99,86%	99,87%	99,87%
ToddlerFountain	152639,74	99,76%	99,84%	99,94%	99,84%	99,85%	99,84%
CatRobot	37798,05	99,42%	99,62%	99,85%	99,74%	99,77%	99,83%
DaylightRoad	112422,11	99,74%	99,83%	99,94%	99,83%	99,86%	99,86%
RollerCoaster	204473,38	99,82%	99,89%	99,97%	99,89%	99,89%	99,89%
TrafficFlow	52781,21	99,55%	99,71%	99,89%	99,77%	99,81%	99,88%
BasketballDrive	43491,76	99,81%	99,87%	99,96%	99,87%	99,88%	99,88%
BQTerrace	10865,76	99,41%	99,60%	99,82%	99,74%	99,74%	99,78%
Cactus	8905,01	99,33%	99,55%	99,81%	99,67%	99,71%	99,75%
Kimono	31632,80	99,76%	99,84%	99,95%	99,84%	99,85%	99,86%
ParkScene	9835,95	99,37%	99,58%	99,82%	99,69%	99,73%	99,79%
Tennis	42405,57	99,80%	99,87%	99,96%	99,87%	99,87%	99,87%
BlueSky	37933,63	99,80%	99,86%	99,94%	99,86%	99,87%	99,87%
InToTree	51826,92	99,83%	99,88%	99,96%	99,88%	99,89%	99,89%
PedestrianArea	29965,42	99,76%	99,84%	99,93%	99,85%	99,87%	99,87%
RushHour	16841,61	99,64%	99,76%	99,90%	99,78%	99,82%	99,85%
Sunflower	47473,85	99,82%	99,87%	99,95%	99,88%	99,89%	99,88%
Tractor	49654,60	99,81%	99,87%	99,95%	99,87%	99,88%	99,87%
Média	63702,47	99,68%	99,79%	99,92%	99,82%	99,84%	99,85%
Desvio Padrão	53964,03	0,17%	0,11%	0,05%	0,07%	0,06%	0,04%

solução altamente eficiente no combate ao alto número de dados lidos na memória externa.

Como os ganhos do *Level C* são muito expressivos, os ganhos provenientes do uso dos cortes fixos e das estratégias de gerenciamento dinâmico ficam

mascaradas no cenário geral. No entanto, o uso do gerenciamento dinâmico da estimação de movimento neste cenário é muito vantajoso, pois mesmo um cenário onde atua o Level C ainda traz um grande consumo de energia. Tendo isto em visto, a inserção do gerenciamento dinâmico se torna altamente vantajosa. Como apresentado nos resultados finais discutidos na primeira parte deste capítulo, o gerenciamento dinâmico pode reduzir, por meio da solução GD MV, em média 47,04% do consumo de energia total em relação a um sistema onde já atuam o *Level C* e o compressor de quadros de referência DDRFC com um impacto negligível em eficiência de codificação. Isso porque o gerenciamento dinâmico da ME proposto neste trabalho explora de forma eficiente diferentes regiões de um mesmo vídeo, alcançando boas relações entre eficiência de codificação e redução de consumo de energia.

7 CONCLUSÕES

Este trabalho apresentou um projeto de memória de busca e um gerenciamento dinâmico de estimação de movimento centrado nessa memória, que permite *power gating* a nível de setores. Para que o projeto de memória fosse possível, este trabalho compreendeu um estudo estatístico sobre a distribuição dos acessos às amostras pelo algoritmo TZS durante a etapa de estimação de movimento. Por meio deste estudo, foi possível concluir que a maioria dos acessos tende a ficar concentrados na região central da área de busca. Inclusive, 95% das amostras mais acessadas se concentram em uma região que corresponde a 66,65% da área de busca. Com os dados obtidos a partir da análise desta distribuição, reduções fixas na janela de busca foram realizadas a fim de verificar o impacto na eficiência de codificação quando algumas das amostras menos acessadas são removidas da área de busca.

Com isso, quatro configurações de tamanho de área de busca foram testadas e comparadas com a área de busca original, com SR [-64, +63]. Estes tamanhos compreenderam a SR [-32, +31], e os CFs 33,35%, 65,60% e 81,36%. Os resultados mostraram que o CF 33,35% apresentou o menor impacto em eficiência de codificação do que os demais. Porém, esta configuração ainda permite um grande volume de comparações e é a que alcança a menor redução em número de amostras acessadas quando comparado com as demais configurações de tamanho de área de busca. Ao mesmo tempo, configurações que apresentaram reduções maiores em número de amostras comparadas obtiveram maiores variações em BD-Rate.

Visando obter a melhor relação entre redução de consumo de energia e eficiência de codificação, este trabalho propôs uma memória dividida em dois setores. O setor mais externo dessa memória pode sofrer *power gating* quando os acessos neste setor se tornam pouco relevantes para a estimação de movimento. Dessa forma, esse trabalho abordou onze modos de gerenciamento dinâmico da estimação de movimento. Estes modos apresentaram diferentes granularidades e estratégias para a tomada de decisão de desligar ou não o setor mais externo.

Como o gerenciamento dinâmico permite a melhor exploração das características de vídeo, as estratégias abordadas conseguiram resultados de

eficiência de codificação próximos dos obtidos pelo corte fixo menor agressivo, o CF 33,35%, com a vantagem alcançar reduções mais significativas em consumo de energia.

Para a obtenção dos resultados de consumo, o projeto de memória e as estratégias de gerenciamento dinâmico foram integrados com o modelo de avaliação de consumo de energia proposto em (AMARAL, 2015). No entanto, como o desligamento de setores implica em redução do consumo estático da memória interna, uma extensão foi feita no modelo para permitir o cálculo desse consumo. Desta forma, foram apresentados e discutidos os resultados de consumo dinâmico, consumo estático e consumo total para os cortes fixos e as estratégias de gerenciamento dinâmico de estimação de movimento abordados. A partir dessas análises, notou-se que quanto menos permissivas as soluções de gerenciamento dinâmico, maior a tendência de um maior impacto em eficiência de codificação e maior a redução energética obtida pela solução. Por outro lado, quanto mais permissivas, menor o impacto em eficiência de codificação e menor a redução de consumo de energia alcançada.

Com as relações entre eficiência de codificação e redução de consumo obtidas, três soluções de gerenciamento de estimação de movimento e dois cortes fixos (CF 33,35% e CF 81,36%) tiveram seus resultados expandidos para um conjunto maior de vídeos. Dentre essas soluções, a GD MV foi a que obteve o melhor resultado médio em BD-Rate, 0,49%, e uma redução de consumo de energia de 43,59% quando comparado com a solução onde a área de busca original é adotada, sendo a estratégia que obteve o melhor balanço entre redução de consumo e impacto na eficiência de codificação.

Além disso, como a memória de busca proposta está inserida em um sistema que conta também com uma estratégia de reuso de dados *Level/C* e o compressor de quadros de referência DDRFC, uma avaliação do quanto a inserção dos cortes fixos e do gerenciamento dinâmico de ME impacta este sistema também foi feita. Com isso, os sistemas com *Level/C*, DDRFC e gerenciamento dinâmico atingiram reduções de consumo de energia superiores a 98,8%, com baixa variação em BD-Rate quando comparados com o sistema onde nenhuma técnica é aplicada.

O projeto de memória de busca apresentado neste trabalho se baseou no comportamento dos acessos do TZS. Essa mesma metodologia pode ser seguida para outros algoritmos de ME, a fim de otimizar o projeto de acordo com o algoritmo

adotado. Novas explorações de soluções de gerenciamento dinâmico de ME também podem ser abordadas.

No entanto, com os resultados obtidos e discutidos neste trabalho, percebe-se que alguns vídeos se comportam melhor com determinada estratégia, apresentando resultados melhores tanto em BD-Rate quanto em redução de consumo de energia do que com outras soluções de cortes fixos ou gerenciamento dinâmico. Assim, uma proposta de um novo controlador pode ser feita, onde este controlador analisa a relação entre *bitrate* e qualidade durante a codificação de um vídeo e, de forma dinâmica, habilita uma das estratégias de corte fixo ou gerenciamento dinâmico abordadas neste trabalho para codificar o próximo conjunto de quadros ou CTUs.

Por fim, considerando o escopo do sistema avaliado como um todo, também podem ser avaliadas outras hierarquias de memórias envolvendo o uso de múltiplas memórias rápidas, inseridas antes e/ou depois do compressor de quadros de referência. Também podem ser avaliados diferentes tipos de reuso de dados e/ou uso de memórias *cache*. Ademais, outra técnica a ser avaliada na redução de largura de banda de memória é manter os 8 bits das amostras dos quadros de referência no DPB, mas utilizar essas amostras com um número menor de bits durante a ME, descartando as informações menos significativas.

Apêndices

APÊNDICE A – Resultados de BD-Rate das soluções avaliadas

Os resultados de BD-Rate se referem aos vídeos (1) *BasketballDrive*, (2) *BQTerrace*, (3) *Cactus*, (4) *Kimono*, (5) *ParkScene* e (6) *Tennis*.

Tabela 1. Resultados de BD-Rate (%) das soluções avaliadas

Solução	Sequência						Média	Mínimo	Máximo	Desvio Padrão
	(1)	(2)	(3)	(4)	(5)	(6)				
CF 33,35%	0,2083	-0,1190	-0,0424	-0,0528	-0,4908	0,1013	-0,0659	-0,4908	0,2083	0,2397
CF 66,65%	1,0799	0,1673	0,4400	0,1902	-0,3358	1,3196	0,4769	-0,3358	1,3196	0,6186
CF 81,36%	4,7787	0,3640	1,1027	2,1037	0,1799	4,9770	2,2510	0,1799	4,9770	2,1453
GD 4Q 1/3	0,3862	0,5259	0,7316	0,1772	0,0367	0,1013	0,3265	0,0367	0,7316	0,2700
GD 4Q 1/5	0,1765	0,0676	0,0554	-0,0613	-0,7439	0,1013	-0,0674	-0,7439	0,1765	0,3403
GD L 1/2	0,3751	0,3137	0,4979	0,2709	-0,0260	0,1770	0,2681	-0,0260	0,4979	0,1795
GD L 1/3	0,2353	0,1221	-0,0349	0,1787	0,4124	0,1013	0,1691	-0,0349	0,4124	0,1497
GD ML 1/3	0,1736	0,1019	0,2612	-0,0182	0,6226	0,4048	0,2577	-0,0182	0,6226	0,2290
GD CTU	0,6303	0,6835	0,1687	0,4045	-0,5254	0,4137	0,2959	-0,5254	0,6835	0,4424
GD CTU OT	0,2904	0,4859	0,2036	0,4361	-0,1472	0,1755	0,2407	-0,1472	0,4859	0,2267
GD MV	0,3551	0,2974	-0,1264	-0,0283	-0,5227	0,1405	0,0193	-0,5227	0,3551	0,3231
GD ESO	0,3767	0,3276	-0,0446	0,0345	-0,1054	0,1615	0,1250	-0,1054	0,3767	0,1979
GD <i>Blur</i>	0,1422	0,5049	0,0702	-0,0329	-0,3488	0,2362	0,0953	-0,3488	0,5049	0,2843
GD JCTU	0,0931	0,5296	-0,0482	0,0290	-0,3280	0,2206	0,0827	-0,3280	0,5296	0,2855

APÊNDICE B – Resultados de consumo de energia das soluções avaliadas

Os resultados de consumo das tabelas se referem aos vídeos (1) *BasketballDrive*, (2) *BQTerrace*, (3) *Cactus*, (4) *Kimono*, (5) *ParkScene* e (6) *Tennis*.

Tabela 1. Resultados de consumo dinâmico (mJ) das soluções avaliadas

Solução	Sequência						Média	Desvio Padrão
	(1)	(2)	(3)	(4)	(5)	(6)		
SR [-64,+63]	79,27	58,71	54,24	71,36	56,64	77,55	66,30	11,11
CF 33,35%	52,47	40,70	36,72	48,06	38,28	51,42	44,61	6,89
CF 66,65%	39,25	37,59	34,14	38,52	35,46	38,47	37,24	2,00
CF 81,36%	16,23	18,50	15,72	16,00	16,40	15,97	16,47	1,02
GD 4Q 1/3	50,54	24,79	21,39	45,97	23,64	51,42	36,29	14,42
GD 4Q 1/5	51,60	26,95	27,49	47,79	28,42	51,42	38,95	12,49
GD L 1/2	49,57	22,72	20,68	41,90	19,04	51,19	34,18	15,02
GD L 1/3	50,94	24,76	23,36	46,08	20,95	51,42	36,25	14,66
GD ML 1/3	50,74	25,42	24,17	45,72	24,30	51,34	36,95	13,64
GD CTU	49,19	25,25	23,02	42,00	23,51	50,28	35,54	13,06
GD CTU OT	48,62	25,20	22,76	41,44	23,38	48,29	34,95	12,53
GD MV	50,05	26,57	23,97	44,12	24,45	51,21	36,73	13,10
GD ESO	49,96	26,48	24,20	43,75	24,67	51,20	36,71	12,97
GD <i>Blur</i>	50,83	28,42	26,24	46,15	26,47	51,29	38,23	12,41
GD JCTU	50,73	28,26	25,88	45,73	27,15	51,29	38,17	12,31

Tabela 2. Reduções de consumo dinâmico (%) das soluções avaliadas

Solução	Sequência						Média	Desvio Padrão
	(1)	(2)	(3)	(4)	(5)	(6)		
SR [-64,+63] (mJ)	79,27	58,71	54,24	71,36	56,64	77,55	66,30	11,11
CF 33,35%	33,81%	30,68%	32,29%	32,64%	32,42%	33,69%	32,59%	1,14%
CF 66,65%	50,49%	35,97%	37,05%	46,02%	37,40%	50,40%	42,89%	6,87%
CF 81,36%	79,52%	68,49%	71,01%	77,57%	71,04%	79,41%	74,51%	4,88%
GD 4Q 1/3	36,24%	57,78%	60,56%	35,57%	58,27%	33,69%	47,02%	13,04%
GD 4Q 1/5	34,90%	54,10%	49,32%	33,03%	49,82%	33,69%	42,48%	9,59%
GD L 1/2	37,46%	61,30%	61,87%	41,29%	66,39%	33,99%	50,38%	14,32%
GD L 1/3	35,74%	57,83%	56,93%	35,43%	63,01%	33,69%	47,11%	13,49%
GD ML 1/3	35,99%	56,70%	55,44%	35,93%	57,10%	33,80%	45,83%	11,64%
GD CTU	37,95%	57,00%	57,56%	41,14%	58,50%	35,17%	47,89%	10,91%
GD CTU OT	38,66%	57,08%	58,05%	41,92%	58,73%	37,73%	48,69%	10,25%
GD MV	36,86%	54,74%	55,82%	38,17%	56,83%	33,97%	46,06%	10,77%
GD ESO	36,97%	54,89%	55,39%	38,69%	56,44%	33,98%	46,06%	10,54%
GD <i>Blur</i>	35,88%	51,59%	51,62%	35,32%	53,28%	33,86%	43,59%	9,43%
GD JCTU	36,00%	51,87%	52,30%	35,92%	52,07%	33,86%	43,67%	9,24%

Tabela 3. Resultados de consumo estático (mJ) das soluções avaliadas

Solução	Sequência						Média	Desvio Padrão
	(1)	(2)	(3)	(4)	(5)	(6)		
SR [-64,+63]	4,79	4,59	4,56	4,82	4,55	4,82	4,69	0,14
CF 33,35%	2,77	2,65	2,63	2,78	2,63	2,79	2,71	0,08
CF 66,65%	1,44	1,39	1,38	1,45	1,38	1,45	1,41	0,04
CF 81,36%	0,72	0,69	0,69	0,72	0,69	0,72	0,70	0,02
GD 4Q 1/3	2,59	1,02	1,03	2,59	1,17	2,79	1,87	0,87
GD 4Q 1/5	2,69	1,23	1,62	2,76	1,61	2,79	2,11	0,70
GD L 1/2	2,51	0,88	0,96	2,25	0,76	2,76	1,69	0,91
GD L 1/3	2,62	1,06	1,21	2,60	0,93	2,79	1,87	0,88
GD ML 1/3	2,60	1,08	1,28	2,57	1,21	2,77	1,92	0,81
GD CTU	2,45	1,04	1,14	2,20	1,10	2,66	1,76	0,75
GD CTU OT	2,45	1,08	1,17	2,23	1,15	2,59	1,78	0,72
GD MV	2,56	1,21	1,29	2,45	1,26	2,76	1,92	0,74
GD ESO	2,55	1,22	1,31	2,43	1,29	2,76	1,93	0,72
GD <i>Blur</i>	2,62	1,41	1,51	2,63	1,45	2,76	2,06	0,67
GD JCTU	2,61	1,38	1,48	2,59	1,51	2,77	2,06	0,66

Tabela 4. Reduções de consumo estático (%) das soluções avaliadas

Solução	Sequência						Média	Desvio Padrão
	(1)	(2)	(3)	(4)	(5)	(6)		
SR [-64,+63] (mJ)	4,79	4,59	4,56	4,82	4,55	4,82	4,69	0,14
CF 33,35%	42,21%	42,19%	42,19%	42,20%	42,19%	42,21%	42,20%	0,01%
CF 66,65%	69,87%	69,76%	69,76%	69,86%	69,75%	69,88%	69,82%	0,06%
CF 81,36%	85,06%	84,89%	84,89%	85,07%	84,88%	85,08%	84,98%	0,10%
GD 4Q 1/3	45,84%	77,78%	77,40%	46,13%	74,30%	42,21%	60,61%	17,50%
GD 4Q 1/5	43,93%	73,28%	64,34%	42,68%	64,68%	42,21%	55,19%	13,80%
GD L 1/2	47,64%	80,91%	78,84%	53,30%	83,18%	42,72%	64,43%	18,48%
GD L 1/3	45,30%	76,86%	73,49%	45,96%	79,50%	42,21%	60,55%	17,74%
GD ML 1/3	45,63%	76,55%	71,96%	46,68%	73,35%	42,45%	59,43%	16,03%
GD CTU	48,89%	77,32%	75,06%	54,38%	75,82%	44,84%	62,72%	14,95%
GD CTU OT	48,86%	76,45%	74,35%	53,73%	74,78%	46,33%	62,41%	14,21%
GD MV	46,58%	73,57%	71,75%	49,08%	72,34%	42,75%	59,35%	14,62%
GD ESO	46,74%	73,46%	71,16%	49,51%	71,60%	42,76%	59,21%	14,28%
GD <i>Blur</i>	45,28%	69,25%	66,89%	45,48%	68,01%	42,67%	56,26%	12,97%
GD JCTU	45,50%	69,82%	67,60%	46,21%	66,76%	42,60%	56,42%	12,85%

Tabela 5. Resultados de consumo total (mJ) das soluções avaliadas

Solução	Sequência						Média	Desvio Padrão
	(1)	(2)	(3)	(4)	(5)	(6)		
SR [-64,+63]	84,06	63,30	58,80	76,17	61,19	82,37	70,98	11,23
CF 33,35%	55,24	43,35	39,36	50,85	40,91	54,21	47,32	6,97
CF 66,65%	40,69	38,98	35,52	39,97	36,83	39,92	38,65	2,04
CF 81,36%	16,95	19,20	16,41	16,72	17,09	16,69	17,18	1,02
GD 4Q 1/3	53,14	25,81	22,42	48,57	24,81	54,21	38,16	15,29
GD 4Q 1/5	54,29	28,18	29,11	50,55	30,03	54,21	41,06	13,18
GD L 1/2	52,08	23,60	21,65	44,15	19,80	53,95	35,87	15,93
GD L 1/3	53,56	25,82	24,57	48,68	21,88	54,21	38,12	15,54
GD ML 1/3	53,35	26,50	25,44	48,29	25,51	54,11	38,87	14,44
GD CTU	51,64	26,29	24,15	44,20	24,60	52,93	37,30	13,80
GD CTU OT	51,07	26,28	23,92	43,67	24,52	50,88	36,73	13,24
GD MV	52,61	27,79	25,25	46,57	25,71	53,97	38,65	13,84
GD ESO	52,51	27,70	25,51	46,18	25,96	53,96	38,64	13,69
GD <i>Blur</i>	53,45	29,83	27,75	48,78	27,92	54,05	40,30	13,07
GD JCTU	53,34	29,65	27,35	48,32	28,66	54,06	40,23	12,96

Tabela 6. Reduções de consumo total (%) das soluções avaliadas

Solução	Sequência						Média	Desvio Padrão
	(1)	(2)	(3)	(4)	(5)	(6)		
SR [-64,+63] (mJ)	84,06	63,30	58,80	76,17	61,19	82,37	70,98	11,23
CF 33,35%	34,29%	31,51%	33,06%	33,25%	33,14%	34,19%	33,24%	1,00%
CF 66,65%	51,60%	38,42%	39,59%	47,53%	39,80%	51,54%	44,75%	6,19%
CF 81,36%	79,84%	69,67%	72,09%	78,05%	72,07%	79,74%	75,24%	4,48%
GD 4Q 1/3	36,79%	59,23%	61,87%	36,24%	59,46%	34,19%	47,96%	13,45%
GD 4Q 1/5	35,41%	55,49%	50,48%	33,64%	50,93%	34,19%	43,36%	9,97%
GD L 1/2	38,04%	62,72%	63,19%	42,05%	67,64%	34,50%	51,36%	14,71%
GD L 1/3	36,29%	59,21%	58,22%	36,09%	64,24%	34,19%	48,04%	13,88%
GD ML 1/3	36,53%	58,14%	56,72%	36,61%	58,31%	34,31%	46,77%	12,04%
GD CTU	38,57%	58,47%	58,92%	41,98%	59,79%	35,74%	48,91%	11,30%
GD CTU OT	39,24%	58,48%	59,31%	42,67%	59,93%	38,23%	49,64%	10,62%
GD MV	37,42%	56,11%	57,05%	38,86%	57,98%	34,48%	46,98%	11,13%
GD ESO	37,53%	56,24%	56,61%	39,38%	57,57%	34,49%	46,97%	10,90%
GD <i>Blur</i>	36,41%	52,87%	52,81%	35,96%	54,37%	34,38%	44,47%	9,77%
GD JCTU	36,54%	53,17%	53,48%	36,57%	53,16%	34,37%	44,55%	9,59%

APÊNDICE C – Resultados finais de consumo de energia das soluções avaliadas

Tabela 1. Resultados finais de consumo dinâmico (mJ)

Sequência	SR [-64,+63]	CF 33,35%	CF 81,36%	GD 4Q 1/5	GD MV	GD L 1/2
CampfireParty	228,96	149,69	52,38	126,01	120,26	82,87
Drums	263,74	176,74	62,74	159,90	145,44	131,74
Tango	298,96	194,51	58,00	194,15	179,34	184,63
ToddlerFountain	337,80	228,72	80,77	228,72	220,81	223,67
CatRobot	198,94	130,97	50,53	90,53	82,00	58,90
DaylightRoad	267,83	176,03	59,20	176,05	148,04	145,79
RollerCoaster	338,23	220,06	62,02	220,06	211,88	207,26
TrafficFlow	215,34	140,77	51,79	111,43	94,83	60,45
BlueSky	71,54	49,84	21,00	48,56	45,40	46,44
InToTree	83,44	57,04	21,03	57,04	54,16	56,18
PedestrianArea	67,14	45,93	18,39	43,09	37,59	37,19
RushHour	55,36	37,68	16,08	33,60	28,41	24,19
Sunflower	81,34	57,06	22,66	53,55	51,38	52,29
Tractor	88,08	62,68	26,09	62,68	58,17	61,35
Média	185,48	123,41	43,05	114,67	105,55	98,07
Desvio Padrão	107,39	69,84	21,28	69,50	65,82	67,26

Tabela 2. Resultados finais de redução consumo dinâmico (%)

Sequência	SR [-64,+63]	CF 33,35%	CF 81,36%	GD 4Q 1/5	GD MV	GD L 1/2
CampfireParty	228,96	34,62%	77,12%	44,97%	47,48%	63,80%
Drums	263,74	32,99%	76,21%	39,37%	44,85%	50,05%
Tango	298,96	34,94%	80,60%	35,06%	40,01%	38,24%
ToddlerFountain	337,80	32,29%	76,09%	32,29%	34,63%	33,79%
CatRobot	198,94	34,17%	74,60%	54,49%	58,78%	70,39%
DaylightRoad	267,83	34,28%	77,90%	34,27%	44,73%	45,56%
RollerCoaster	338,23	34,94%	81,66%	34,94%	37,36%	38,72%
TrafficFlow	215,34	34,63%	75,95%	48,25%	55,96%	71,93%
BlueSky	71,54	30,33%	70,65%	32,12%	36,54%	35,10%
InToTree	83,44	31,64%	74,80%	31,64%	35,08%	32,67%
PedestrianArea	67,14	31,60%	72,61%	35,82%	44,01%	44,62%
RushHour	55,36	31,93%	70,96%	39,30%	48,69%	56,31%
Sunflower	81,34	29,85%	72,14%	34,16%	36,83%	35,72%
Tractor	88,08	28,84%	70,38%	28,84%	33,96%	30,35%
Média	185,48	32,64%	75,12%	37,54%	42,78%	46,23%
Desvio Padrão	107,39	2,04%	3,54%	7,18%	7,90%	14,19%

Tabela 3. Resultados finais de consumo estático (mJ)

Sequência	SR [-64,+63]	CF 33,35%	CF 81,36%	GD 4Q 1/5	GD MV	GD L 1/2
CampfireParty	18,44	10,66	2,78	7,73	6,89	4,50
Drums	19,08	11,03	2,86	9,45	8,16	6,84
Tango	20,22	11,69	3,03	11,67	10,33	10,85
ToddlerFountain	20,80	12,02	3,10	12,02	11,27	11,57
CatRobot	18,18	10,51	2,75	6,13	5,15	2,94
DaylightRoad	18,80	10,86	2,82	10,85	8,00	8,11
RollerCoaster	20,64	11,93	3,08	11,93	11,10	10,75
TrafficFlow	18,23	10,54	2,75	7,61	5,56	2,90
BlueSky	4,76	2,75	0,71	2,62	2,51	2,45
InToTree	4,77	2,76	0,71	2,76	2,73	2,67
PedestrianArea	4,68	2,71	0,70	2,41	1,99	1,90
RushHour	4,62	2,67	0,70	2,22	1,83	1,38
Sunflower	4,91	2,84	0,73	2,52	2,55	2,41
Tractor	4,97	2,87	0,74	2,87	2,75	2,77
Média	13,08	7,56	1,96	6,63	5,77	5,15
Desvio Padrão	7,50	4,33	1,13	4,02	3,52	3,70

Tabela 4. Resultados finais de redução consumo estático (%)

Sequência	SR [-64,+63]	CF 33,35%	CF 81,36%	GD 4Q 1/5	GD MV	GD L 1/2
CampfireParty	18,44	42,20%	84,93%	58,07%	62,65%	75,59%
Drums	19,08	42,20%	85,03%	50,47%	57,23%	64,14%
Tango	20,22	42,21%	85,02%	42,31%	48,92%	46,36%
ToddlerFountain	20,80	42,20%	85,10%	42,20%	45,81%	44,36%
CatRobot	18,18	42,19%	84,87%	66,28%	71,67%	83,84%
DaylightRoad	18,80	42,20%	84,98%	42,28%	57,43%	56,85%
RollerCoaster	20,64	42,22%	85,09%	42,22%	46,20%	47,93%
TrafficFlow	18,23	42,20%	84,89%	58,26%	69,51%	84,08%
BlueSky	4,76	42,21%	85,03%	44,85%	47,13%	48,37%
InToTree	4,77	42,40%	85,06%	42,22%	42,81%	44,13%
PedestrianArea	4,68	41,32%	84,99%	48,59%	57,50%	59,35%
RushHour	4,62	40,50%	84,93%	51,87%	60,35%	70,09%
Sunflower	4,91	44,04%	85,13%	48,71%	48,00%	50,85%
Tractor	4,97	44,72%	85,16%	42,21%	44,64%	44,26%
Média	13,08	42,34%	85,02%	48,61%	54,28%	58,59%
Desvio Padrão	7,50	1,01%	0,09%	7,68%	9,41%	14,64%

Tabela 5. Resultados finais de consumo total (mJ)

Sequência	SR [-64,+63]	CF 33,35%	CF 81,36%	GD 4Q 1/5	GD MV	GD L 1/2
CampfireParty	247,40	160,35	55,16	133,74	127,14	87,37
Drums	282,82	187,77	65,60	169,35	153,60	138,59
Tango	319,18	206,20	61,03	205,81	189,67	195,48
ToddlerFountain	358,60	240,74	83,87	240,74	232,08	235,24
CatRobot	217,11	141,48	53,28	96,66	85,86	61,84
DaylightRoad	286,62	186,89	62,02	186,90	156,04	153,91
RollerCoaster	358,86	231,99	65,10	231,99	222,98	218,01
TrafficFlow	233,56	151,31	54,54	119,04	100,38	63,35
BlueSky	76,30	52,59	21,71	51,19	47,92	48,89
InToTree	88,21	59,80	21,74	59,80	56,89	58,84
PedestrianArea	71,83	48,64	19,10	45,50	39,58	39,09
RushHour	59,98	40,35	16,77	35,83	30,24	25,57
Sunflower	86,25	59,90	23,39	56,07	53,94	54,70
Tractor	93,05	65,55	26,83	65,55	60,92	64,12
Média	198,56	130,97	45,01	121,30	111,23	103,21
Desvio Padrão	114,62	74,00	22,36	73,46	69,35	70,95

Tabela 6. Resultados finais de redução consumo total (%)

Sequência	SR [-64,+63]	CF 33,35%	CF 81,36%	GD 4Q 1/5	GD MV	GD L 1/2
CampfireParty	247,40	35,19%	77,70%	45,94%	48,61%	64,68%
Drums	282,82	33,61%	76,81%	40,12%	45,69%	51,00%
Tango	319,18	35,40%	80,88%	35,52%	40,57%	38,76%
ToddlerFountain	358,60	32,86%	76,61%	32,86%	35,28%	34,40%
CatRobot	217,11	34,84%	75,46%	55,48%	60,45%	71,52%
DaylightRoad	286,62	34,80%	78,36%	34,79%	45,56%	46,30%
RollerCoaster	358,86	35,36%	81,86%	35,36%	37,86%	39,25%
TrafficFlow	233,56	35,22%	76,65%	49,03%	57,02%	72,88%
BlueSky	76,30	31,07%	71,55%	32,92%	37,20%	35,92%
InToTree	88,21	32,21%	75,36%	32,21%	35,50%	33,29%
PedestrianArea	71,83	32,29%	73,41%	36,65%	44,89%	45,58%
RushHour	59,98	32,72%	72,03%	40,27%	49,58%	57,38%
Sunflower	86,25	30,55%	72,88%	34,99%	37,47%	36,58%
Tractor	93,05	29,55%	71,17%	29,55%	34,53%	31,09%
Média	198,56	33,26%	75,77%	38,26%	43,59%	47,04%
Desvio Padrão	114,62	1,96%	3,32%	7,29%	8,19%	14,33%

REFERÊNCIAS

AFONSO, V. et al. Energy-Aware Motion and Disparity Estimation System for 3D-HEVC with Run-Time Adaptive Memory Hierarchy. **IEEE Transactions on Circuits and Systems for Video Technology**, 2018. (Artigo submetido para publicação)

AMARAL, L. **Modelagem SystemC para Avaliação Energética da Comunicação com a Memória em Codificadores de Vídeo**. Universidade Federal de Pelotas. Pelotas, p. 73. 2015.

BOSSSEN, F. et al. HEVC Complexity and Implementation Analysis. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 22, n. 12, p. 1685-1696, 2012.

CHANG, Y.-T.; CHUNG, W.-H. **A dynamic search range algorithm for H.264/AVC full-search motion estimation**. IEEE Asia Pacific Conference on Circuits and Systems (APCCAS). [S.l.]: [s.n.]. 2010. p. 124 - 127.

CHEN, C.-Y. et al. Level C+ data reuse scheme for motion estimation with corresponding coding orders. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 16, n. 4, p. 553-558, April 2006.

CHENG, C.-C.; TSENG, P.-C.; CHEN, L.-G. Multimode Embedded Compression Codec Engine for Power-Aware Video Coding System. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 19, n. 2, p. 141-150, 2009.

CHIEN, W.-D.; LIAO, K.-Y.; YANG, J.-F. **Enhanced AMVP mechanism based adaptive motion search range decision algorithm for fast HEVC coding**. IEEE International Conference on Image Processing (ICIP). [S.l.]: [s.n.]. 2014.

CISCO. **Cisco Visual Networking Index: Forecast and Methodology, 2016-2021 White Paper**. Cisco. [S.l.]. 2017.

DAI, W. et al. **Adaptive search range algorithm based on Cauchy distribution**. IEEE Visual Communications and Image Processing (VCIP). [S.l.]: [s.n.]. 2012.

DU, L. et al. **Linear adaptive search range model for uni-prediction and motion analysis for bi-prediction in HEVC**. IEEE International Conference on Image Processing (ICIP). [S.l.]: [s.n.]. 2014.

FACEBOOK. Sight, Sound and Mobilization: A Facebook IQ Mobile Video Study. **Facebook**, 2017. Disponível em: <<https://www.facebook.com/iq/articles/sight-sound-and-mobilization>>. Acesso em: 6 Fevereiro 2018.

GRELLERT, M. et al. **A Multilevel Data Reuse Scheme for Motion Estimation and its VLSI Design**. International Symposium on Circuits and Systems (ISCAS). Rio de Janeiro: [s.n.]. 2011.

GRELLERT, M.; BAMPI, S.; ZATT, B. **Complexity-scalable HEVC encoding**. Picture Coding Symposium (PCS). [S.I.]: [s.n.]. 2016.

HM. HEVC Test Model (HM), 2016. Disponível em: <https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.6/>. Acesso em: 24 Março 2018.

INSTAGRAM. Imagem, som e mobilização. **Instagram**, 2017. Disponível em: <<https://business.instagram.com/blog/sight-sound-and-mobilization>>. Acesso em: 6 Fevereiro 2018.

ITU-T. **International Telecommunication Union. Recommendation H.264 (01/12): Advanced video coding for generic audiovisual services**. [S.I.]. 2012.

ITU-T. **Recommendation H.265: High Efficiency Video Coding, Audiovisual and Multimedia Systems**. [S.I.]. 2013.

JAKUBOWSKI, M.; PASTUSZAK, G. **Data Reuse in Two-Level Hierarchical Motion Estimation for High Resolution Video Coding**. International Signal Processing and Multimedia Applications. Atena: [s.n.]. 2010. p. 159-162.

JASUJA, N. SRAM vs DRAM. **Diffen**, 2018. Disponível em: <https://www.diffen.com/difference/Dynamic_random-access_memory_vs_Static_random-access_memory>. Acesso em: 1 Março 2018.

Jl, X. et al. Computation-constrained dynamic search range control for real-time video encoder. **Signal Processing: Image Communication**, 2015. 134-150.

JIA, L. et al. **A diamond search windowbased adaptive search range algorithm**. IEEE International Conference on Multimedia and Expo Workshops (ICMEW). [S.I.]: [s.n.]. 2013. p. 1- 4.

JUNG, J.; KIM, J.; KYUNG, C.-M. A Dynamic Search Range Algorithm for Stabilized Reduction of Memory Traffic in Video Encoder. **IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)**, p. 1041 - 1046, 2010.

KUHN, P. M. **Algorithms, Complexity Analysis and VLSI Architectures for MPEG-4 Motion Estimation**. [S.I.]: Kluwer Academic Publisher, 1999.

LABS, H. CACTI. **HP Labs: CACTI**, 2017. Disponível em: <<http://www.hpl.hp.com/research/cacti/>>. Acesso em: 2017.

LI, Y. et al. **An adaptive search range method for HEVC with the k-nearest neighbor algorithm**. Visual Communications and Image Processing (VCIP). [S.I.]: [s.n.]. 2015.

MICRON. Micron MT46H64M16LF: 1 Gb DDR SDRAM, 2017. Disponível em: <<http://www.micron.com>>. Acesso em: 2017.

PESQUET-POPESCU, B.; CAGNAZZO, M.; DUFAUX, F. Chapter 2 - Motion Estimation—A Video Coding Viewpoint. In: DUFAUX, F.; THEODORIDIS, S. **Academic Press Library in Signal Processing**. [S.l.]: Elsevier, v. 5, 2014. Cap. 2, p. 27-92.

PORTO, M. **Desenvolvimento Algorítmico e Arquitetural para a Estimação de Movimento na Compressão de Vídeo de Alta Definição**. Universidade Federal do Rio Grande do Sul. Porto Alegre. 2012.

POVALA, G. **Redução de Acesso à Memória Externa na Codificação de Vídeos UHD**. Universidade Federal de Pelotas. Pelotas, p. 84. 2015.

RICHARDSON, I. E. **Video Codec Design: Developing Image and Video Compression Systems**. [S.l.]: John Wiley & Sons, 2002.

RICHARDSON, I. E. **The H.264 Advanced Video Compression Standard**. 2. ed. [S.l.]: John Wiley & Sons, LTD, 2010. 314 p.

SAMPAIO, F. et al. **dSVM: Energy-efficient distributed Scratchpad Video Memory Architecture for the next-generation High Efficiency Video Coding**. Design, Automation and Test in Europe Conference and Exhibition. Dresden: [s.n.]. 2014. p. 1-6.

SAPONARA, S. et al. Dynamic control of motion estimation search parameters for low complex H.264 video coding. **IEEE Transactions on Consumer Electronics**, p. 232 - 239, 2006.

SAPONARA, S.; FANUCCI, L. **Data-adaptive motion estimation algorithm and VLSI architecture design for low-power video systems**. IEE Proceedings - Computers and Digital Techniques. [S.l.]: [s.n.]. 2004. p. 51 - 59.

SHARMAN, K.; KARSTEN, S. **Common test conditions**. JCT-VC. Genebra. 2017. (JCTVC-Z1100).

SILVEIRA, D. **Algoritmos e Arquiteturas de Hardware para a Compressão de Quadros de Referência em Codificadores de Vídeo Digitais**. Universidade Federal de Pelotas. Pelotas, p. 110. 2015.

SILVEIRA, D. et al. Efficient reference frame compression scheme for video coding systems: algorithm and VLSI design. **Journal of Real-Time Image Processing**, Dezembro 2015.

SONG, T. et al. **Adaptive Search Range Motion Estimation Algorithm for H.264/AVC**. IEEE International Symposium on Circuits and Systems. [S.l.]: [s.n.]. 2007. p. 3956 - 3959.

SULLIVAN, G. J. et al. Overview of the High Efficiency Video Coding (HEVC) Standard. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 22, n. 12, p. 1649-1668, December 2012.

SYSTEMC. **Accellera Systems Initiative**, 2018. Disponível em: <<http://accellera.org/community/systemc/>>. Acesso em: 31 Janeiro 2018.

SZE, V.; BUDAGAVI, M.; SULLIVAN, G. J. **High Efficiency Video Coding (HEVC)**. [S.l.]: Springer International Publishing, 2014.

TUAN, J.-C.; CHANG, T.-S.; JEN, C.-W. On the Data Reuse and Memory Bandwidth Analysis for Full-Search Block-Matching VLSI Architecture. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 12, n. 1, p. 61-72, January 2002.

XIU-LI, T.; SHENG-KUI, D.; CAN-HUI, C. **An Analysis of TZSearch Algorithm in JMVC**. International Conference on Green Circuits and Systems. Shanghai: [s.n.]. 2010. p. 516 - 520.

ZATT, B. et al. **Run-Time Adaptive Energy-Aware Motion and Disparity Estimation in Multiview Video Coding**. Design Automation Conference. San Diego: [s.n.]. 2011. p. 1026-1031.