

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



**Dissertação**

**Exploração das Distorções da Projeção ERP para Redução de Complexidade  
da Codificação de Vídeos Omnidirecionais**

**Iago Coelho Storch**

**Pelotas, 2020**

**Iago Coelho Storch**

**Exploração das Distorções da Projeção ERP para Redução de Complexidade  
da Codificação de Vídeos Omnidirecionais**

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. Daniel Munari Vilchez Palomino  
Coorientadores: Prof. Dr. Bruno Zatt  
Prof. Dr. Luciano Volcan Agostini

Pelotas, 2020

Universidade Federal de Pelotas / Sistema de Bibliotecas  
Catalogação na Publicação

S884e Storch, Iago Coelho

Exploração das distorções da projeção ERP para  
redução de complexidade da codificação de vídeos  
omnidirecionais / Iago Coelho Storch ; Daniel Munari  
Vilchez Palomino, orientador ; Bruno Zatt, Luciano Volcan  
Agostini, coorientadores. — Pelotas, 2020.

175 f. : il.

Dissertação (Mestrado) — Programa de Pós-Graduação  
em Computação, Centro de Desenvolvimento Tecnológico,  
Universidade Federal de Pelotas, 2020.

1. Vídeos omnidirecionais. 2. Redução de complexidade.  
3. Projeção equirretangular. 4. Codificação adaptativa ao  
conteúdo. I. Palomino, Daniel Munari Vilchez, orient. II.  
Zatt, Bruno, coorient. III. Agostini, Luciano Volcan, coorient.  
IV. Título.

CDD : 005

**Iago Coelho Storch**

**Exploração das Distorções da Projeção ERP para Redução de Complexidade  
da Codificação de Vídeos Omnidirecionais**

Dissertação aprovada, como requisito parcial, para obtenção do título de Mestre em Ciência da Computação no Programa de Pós-Graduação em Computação da Universidade Federal de Pelotas.

**Data da Defesa:** 7 de fevereiro de 2020

**Banca Examinadora:**

Prof. Dr. Daniel Palomino (orientador, PPGC/UFPel)

Prof. Dr. Cláudio Diniz (UCPel)

Prof. Dr. Marcelo Porto (PPGC/UFPel)

Prof. Dr. Marilton Aguiar (PPGC/UFPel)

Dedico este trabalho aos meus pais.

## AGRADECIMENTOS

A minha escolha de realizar mestrado se deu no ano de 2018. Como se o mestrado já não representasse um desafio por si só, eu ainda decidi que iria concluir-lo em apenas um ano. Como foi alertado pelos meus orientadores na época, "realizar o mestrado na metade do tempo não representa fazer meio trabalho, e sim se dedicar em dobro". Eles não poderiam estar mais certos.

Dessa forma, o desenvolvimento do trabalho que culminou nesta dissertação teve início tão logo quanto o início do ano de 2019. Nestes doze meses houveram inúmeros momentos frustrantes onde eu achei que não seria possível, e foi preciso muita força de vontade para seguir tocando um trabalho que por vezes se mostrava interminável. Essa força de vontade, é claro, só se manifestou devido ao apoio de diversas pessoas que acreditaram em mim e acompanharam minha caminhada, e estas merecem ser citadas neste trabalho.

Primeiramente gostaria de agradecer aos meus pais, Ingomar e Maria Eneida. Eu me sinto uma pessoa extremamente privilegiada por ter tido pais que fizeram todo o possível para me apoiar, tanto financeiramente quanto psicologicamente. Obrigado por me apoiarem quando eu decidi fazer pós-graduação e por permitirem que eu me dedicasse exclusivamente à Universidade. Obrigado por, mesmo sem saber ao certo do que se tratava o meu trabalho, me transmitirem a confiança necessária para seguir desenvolvendo-o, e obrigado por serem tão compreensíveis comigo nos inúmeros momentos de estresse. Parte desta conquista também é deles, e espero que um dia eu seja capaz de retribuir tudo o que fizeram por mim.

A minha namorada Laura também merece ser citada. É preciso muito amor para viajar tanto para ver alguém, e eu me sinto especial por ter encontrado alguém que faz isso por mim. Obrigado por, mesmo a distância, me apoiar e dar energia para me manter motivado, por me confortar nos momentos de angústia, e também por ser tão compreensiva nos meus momentos de ausência. Tuas demonstrações diárias de afeto foram fundamentais para me manter motivado.

Agradeço também aos professores Bruno Zatt, Daniel Palomino, Guilherme Corrêa, Luciano Agostini e Marcelo Porto por todos os valiosos ensinamentos que contribuíram para minha formação como pesquisador e pelo empenho em manter uma infraestrutura adequada para que eu pudesse desenvolver o meu trabalho. Dentre estes, o Daniel e o Marcelo ainda merecem destaque: o primeiro por ter sido meu orientador desde a iniciação científica em 2015 e ter contribuído em todas as etapas da minha trajetória acadêmica, e o segundo por ter visto meu potencial como pesquisador e incentivado que eu entrasse em um grupo de pesquisa ainda nos primeiros semestres da graduação.

Além destes, os colegas de laboratório com quem compartilhei a maior parte dos

meus dias também têm uma participação neste trabalho. Um agradecimento especial ao Douglas, Paulo e Thiago pela disponibilidade de discutir ideias e sanar dúvidas a qualquer instante, pela companhia em todas as refeições no RU e pelos momentos de descontração durante o dia. Estes são alguns dos colegas de laboratório que se tornaram amigos para a vida.

Também gostaria de agradecer a todos os cidadãos brasileiros que financiaram o meu curso de mestrado através da universidade pública e da minha bolsa de mestrado. Vou me esforçar para retribuir à população por esta grande oportunidade.

Por fim, agradeço a todos que de forma direta ou indireta, contribuíram para a realização deste trabalho e não foram mencionados anteriormente.

## RESUMO

STORCH, Iago Coelho. **Exploração das Distorções da Projeção ERP para Redução de Complexidade da Codificação de Vídeos Omnidirecionais**. 2020. 175 f. Dissertação (Mestrado em Ciência da Computação) – Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2020.

A projeção equirretangular (ERP) é uma etapa de pré-processamento que permite que vídeos omnidirecionais sejam codificados através de padrões de codificação de vídeo convencionais. Embora ela possibilite a codificação destes vídeos, a projeção ERP gera distorções no vídeo que se manifestam através de esticamentos horizontais nas texturas de algumas regiões do quadro, e estes esticamentos causam comportamentos característicos nas ferramentas de codificação que não são observados durante a codificação de vídeos convencionais. Sendo assim, este trabalho explora as distorções causadas pela projeção ERP para propor um conjunto de técnicas de redução de complexidade para a codificação de vídeos omnidirecionais, tanto durante a predição interquadros quanto intraquadro. As técnicas aplicadas à predição interquadros são denominadas *Early Skip*, *Reduced SR* e *Reduced FME*. A técnica *Early Skip* consiste em fazer uma decisão prematura pelo modo de codificação *skip* de acordo com a homogeneidade do bloco, a técnica *Reduced SR* consiste em limitar horizontalmente a área de busca da estimativa de movimento, e a técnica *Reduced FME* reduz a precisão horizontal da estimativa de movimento fracionária. Quando as três são utilizadas simultaneamente, é possível reduzir a complexidade do codificador em 22,84%, em média, com um pequeno impacto na eficiência de codificação. Já as técnicas aplicadas à predição intraquadro são denominadas *Reduced Intra Sizes* e *Reduced Intra Modes*. A técnica *Reduced Intra Sizes* consiste em descartar a avaliação de alguns tamanhos de bloco e a técnica *Reduced Intra Modes* foi incorporada de outro trabalho, e baseia-se em priorizar os modos de predição intraquadro horizontais. Quando estas duas técnicas são utilizadas simultaneamente, é possível atingir uma redução de complexidade média de 23,09% sem interferir de forma significativa na eficiência de codificação. Todas as técnicas apresentadas – tanto interquadros quanto intraquadro – avaliam a distorção do vídeo para tomar suas decisões. As técnicas desenvolvidas são competitivas com trabalhos da literatura e demonstram que explorar as distorções causadas pela projeção ERP é uma boa alternativa para reduzir a complexidade na codificação de vídeos omnidirecionais sem penalizar a eficiência de codificação significativamente.

**Palavras-Chave:** vídeos omnidirecionais; redução de complexidade; projeção equirretangular; codificação adaptativa ao conteúdo

## ABSTRACT

STORCH, Iago Coelho. **Exploration of ERP Projection Distortions Aiming Omnidirectional Video Coding Complexity Reduction**. 2020. 175 f. Dissertação (Mestrado em Ciência da Computação) – Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2020.

The equirectangular projection (ERP) is a pre-processing step that allows omnidirectional videos to be encoded through conventional video coding standards. Although it enables the encoding of such videos, ERP projection inserts distortions in the video as horizontal stretching in the textures depending on the frame region, and such distortions introduce specific behaviors in video coding tools that do not occur when encoding conventional videos. Therefore, this work explores the distortions caused by equirectangular projection (ERP) to propose a set of complexity reduction techniques for the encoding of omnidirectional video, which are applied to both interframes and intraframe prediction. The complexity reduction techniques applied to interframe prediction are named *Early Skip*, *Reduced SR* and *Reduced FME*. The *Early Skip* technique is based on performing an early decision to employ the skip prediction mode based on the block homogeneity, the *Reduced SR* technique consists on reducing the width of the search range during motion estimation, and *Reduced FME* technique is based on reducing the horizontal precision of fractional motion estimation. When these three techniques are employed together, it is possible to achieve, on average, 22.84% of complexity reduction with small coding efficiency penalties. The complexity reduction techniques designed to accelerate the intraframe prediction are named *Reduced Intra Sizes* and *Reduced Intra Modes*. *Reduced Intra Sizes* technique discards the evaluation of unlikely block sizes and *Reduced Intra Modes* technique is inherited from another work, and consists on prioritizing the selection of horizontal intra prediction modes. When both intraprediction techniques are employed together, an average complexity reduction 23.09% is observed whereas the coding efficiency is not significantly changed. All designed techniques – both for interframes and intraframe prediction – exploit the ERP distortion throughout the frame to accelerate the encoding properly. In addition, the designed techniques present results competitive with related works and show that it is advantageous to explore the ERP projection distortions to achieve complexity reduction without harming coding efficiency significantly.

**Keywords:** omnidirectional video; complexity reduction; equirectangular projection; content-adaptive encoding

## LISTA DE FIGURAS

Figura 1	Vista frontal e traseira do primeiro quadro do vídeo <i>PoleVault</i> . . . . .	23
Figura 2	Fluxo de processamento de vídeos omnidirecionais . . . . .	25
Figura 3	Exemplos de câmeras omnidirecionais. (a) GoPro Omni, (b) Samsung Gear 360, (c) LG 360 CAM . . . . .	26
Figura 4	Primeiro quadro do vídeo <i>PoleVault</i> na projeção ERP . . . . .	28
Figura 5	Exemplo de extração de <i>viewport</i> (YU; LAKSHMAN; GIROD, 2015)	29
Figura 6	Exemplos de HMDs. (a) Oculus Rift, (b) HTC Vive, (c) Samsung Gear VR . . . . .	30
Figura 7	Diagrama de blocos de um codificador de vídeo genérico (CORRÊA, 2018) . . . . .	32
Figura 8	Blocos em ordem raster sobre o primeiro quadro do vídeo <i>PoleVault</i>	33
Figura 9	Exemplo de funcionamento da predição intraquadro. (a) bloco antes da predição, (b) bloco após a predição . . . . .	34
Figura 10	Exemplo de funcionamento da predição interquadros . . . . .	36
Figura 11	Subdivisão de uma CTU em várias CUs. Adaptado de (CORREA et al., 2015) . . . . .	40
Figura 12	Modos de particionamento de CUs em PUs. Adaptado de (SULLIVAN et al., 2012) . . . . .	41
Figura 13	Modos de predição intraquadro (SULLIVAN et al., 2012) . . . . .	42
Figura 14	Fluxo de Processamento da Predição Intraquadro . . . . .	43
Figura 15	PU atual e vizinhas na derivação dos MPMs . . . . .	44
Figura 16	Fluxograma do algoritmo <i>Test Zone Search</i> . . . . .	45
Figura 17	Padrão de busca da Busca Inicial no Tzs. (a) Diamante e (b) quadrada. Adaptado de (GONÇALVES, 2018) . . . . .	46
Figura 18	Funcionamento da estimativa de movimento fracionária . . . . .	48
Figura 19	Exemplo do primeiro quadro de <i>PoleVault</i> dividido em três e cinco bandas . . . . .	66
Figura 20	Taxa de ocorrência do modo <i>skip</i> para vídeos convencionais. (a) CUs $8 \times 8$ , (b) CUs $16 \times 16$ , (c) CUs $32 \times 32$ , (d) CUs $64 \times 64$ e (e) todos tamanhos ponderados . . . . .	69
Figura 21	Taxa de ocorrência do modo <i>skip</i> para vídeos omnidirecionais. (a) CUs $8 \times 8$ , (b) CUs $16 \times 16$ , (c) CUs $32 \times 32$ , (d) CUs $64 \times 64$ e (e) todos tamanhos ponderados . . . . .	70
Figura 22	Distribuição acumulada de <i>skips</i> de acordo com a variância de CUs $64 \times 64$ em vídeos convencionais, com faixa de variância reduzida .	73

Figura 23	Distribuição acumulada de <i>skips</i> de acordo com a variância de CUs $64 \times 64$ em vídeos omnidirecionais, com faixa de variância reduzida	73
Figura 24	Distribuição acumulada de <i>skips</i> de acordo com a variância de CUs $64 \times 64$ em vídeos convencionais, com faixa de variância total . . . . .	75
Figura 25	Distribuição acumulada de <i>skips</i> de acordo com a variância de CUs $64 \times 64$ em vídeos omnidirecionais, com faixa de variância total . . . . .	75
Figura 26	Distribuição de inclinação dos vetores de movimento da Busca Raster para vídeos convencionais . . . . .	78
Figura 27	Distribuição de inclinação dos vetores de movimento da Busca Raster para vídeos omnidirecionais . . . . .	78
Figura 28	Mapeamento de uma amostra unitária da superfície esférica para o plano. Adaptado de (LUZ, 2017) . . . . .	83
Figura 29	Distribuição dos tamanhos de PU por linha de CTUs para vídeos convencionais. Adaptado de (STORCH et al., 2019) . . . . .	88
Figura 30	Distribuição dos tamanhos de PU por linha de CTUs para vídeos omnidirecionais. Adaptado de (STORCH et al., 2019) . . . . .	88
Figura 31	Visão em alto nível das técnicas desenvolvidas junto ao codificador HEVC . . . . .	91
Figura 32	Algoritmo da técnica <i>Early Skip</i> . . . . .	96
Figura 33	Algoritmo da técnica <i>Reduced SR</i> . . . . .	98
Figura 34	Algoritmo da técnica <i>Reduced FME</i> . . . . .	100
Figura 35	Algoritmo da técnica <i>Reduced Intra Sizes</i> . . . . .	103
Figura 36	Custo RD da PU atual e das PUs em três particionamentos distintos para o quadro de referência . . . . .	105
Figura 37	Diagrama em alto nível da relação entre as técnicas desenvolvidas	106

## LISTA DE TABELAS

Tabela 1	Vídeos omnidirecionais utilizados durante as avaliações e respectivas resoluções . . . . .	63
Tabela 2	Vídeos convencionais utilizados durante as avaliações . . . . .	64
Tabela 3	Faixa angular de cada classe de vetor de movimento . . . . .	77
Tabela 4	Média e desvio padrão das componentes X e Y dos vetores de movimento da Busca Raster . . . . .	79
Tabela 5	Redução de complexidade e eficiência de codificação na alteração da área de Busca Raster nas bandas polares . . . . .	81
Tabela 6	Redução de complexidade e eficiência de codificação na limitação de precisão da FME . . . . .	85
Tabela 7	Variâncias de corte para <i>cutoff</i> igual a 0,45 e diferentes resoluções, tamanhos de bloco, bandas e QPs . . . . .	95
Tabela 8	Sequências usadas para a obtenção dos resultados experimentais .	109
Tabela 9	Redução de complexidade e variação de eficiência de codificação da técnica <i>Early Skip</i> . . . . .	112
Tabela 10	Redução de complexidade e variação de eficiência de codificação da técnica <i>Reduced SR</i> . . . . .	115
Tabela 11	Redução de complexidade e variação de eficiência de codificação da técnica <i>Reduced FME</i> . . . . .	117
Tabela 12	Redução de complexidade e variação de eficiência de codificação da técnica <i>InterTech</i> . . . . .	119
Tabela 13	Redução de complexidade e variação na eficiência de codificação dos trabalhos (RAY; JUNG; LARABI, 2018) e (LIU; SONG; ZHANG, 2019) . . . . .	121
Tabela 14	Redução de complexidade e variação de eficiência de codificação da técnica <i>Reduced Intra Sizes</i> . . . . .	123
Tabela 15	Redução de complexidade e variação de eficiência de codificação da técnica <i>IntraTech</i> . . . . .	125
Tabela 16	Redução de complexidade e variação na eficiência de codificação dos trabalhos (WANG et al., 2017) e (ZHANG et al., 2019) . . . . .	127

## LISTA DE ABREVIATURAS E SIGLAS

2D	Bidimensional
3D	Tridimensional
AMP	<i>Asymmetric motion partitioning</i>
AVC	<i>Advanced Video Coding</i>
BD-BR	<i>Bjontegaard Delta Bitrate</i>
BMA	<i>Block matching algorithm</i>
Cb	Crominância azul
Cr	Crominância vermelha
CTCs	Condições Comuns de Teste para Vídeos Convencionais
CTCs360	Condições Comuns de Teste para Vídeos Omnidirecionais
CTU	<i>Coding tree unit</i>
CU	<i>Coding unit</i>
ERP	Projeção equirretangular
FME	<i>Fractional motion estimation</i>
HEVC	<i>High Efficiency Video Coding</i>
HM	<i>HEVC Test Model</i>
HMD	<i>Head-mounted display</i>
HTTP	<i>Hypertext Transfer Protocol</i>
JVET	<i>Joint Video Experts Team</i>
KNN	<i>K-nearest neighbors</i>
MAD	<i>Mean of the absolute difference</i>
MCP	<i>Motion-compensated prediction</i>
MPEG	<i>Moving Pictures Experts Group</i>
MPM	<i>Most Probable Mode</i>
MSE	<i>Mean squared error</i>
MV	<i>Motion vector</i>

OMAF	<i>Omnidirectional MediA Format</i>
PU	<i>Prediction unit</i>
PSNR	<i>Peak Signal-to-Noise Ratio</i>
QP	<i>Quantization parameter</i>
QT	<i>Quadtree</i>
RCR	Redução de Complexidade Relativa
RDO	<i>Rate-Distortion Optimization</i>
RMD	<i>Rough Mode Decision</i>
RQT	<i>Residual quadtree</i>
SAD	<i>Sum of Absolute Differences</i>
SATD	<i>Sum of Absolute Transformed Differences</i>
TU	<i>Transform unit</i>
TZS	<i>Test Zone Search</i>
VR	<i>Virtual reality</i>
VVC	<i>Versatile Video Coding</i>
WS-PSNR	<i>Weighted-to-Spherically Uniform PSNR</i>
Y	Luminância

# SUMÁRIO

<b>1 INTRODUÇÃO . . . . .</b>	17
<b>1.1 Organização . . . . .</b>	19
<b>2 VÍDEOS OMNIDIRECIONAIS, SUAS CARACTERÍSTICAS E APLICAÇÕES . . . . .</b>	21
<b>2.1 Características de Vídeos Digitais em Geral . . . . .</b>	21
<b>2.2 Características e Aplicações de Vídeos Omnidirecionais . . . . .</b>	23
<b>2.3 Fluxo de Processamento de Vídeos Omnidirecionais . . . . .</b>	24
<b>2.3.1 Captura e Costura . . . . .</b>	25
<b>2.3.2 Projeção Equirretangular - ERP . . . . .</b>	27
<b>2.3.3 Codificação, transmissão/armazenamento e decodificação . . . . .</b>	28
<b>2.3.4 Renderização, Seleção de <i>Viewport</i> e Reprodução . . . . .</b>	29
<b>2.4 Considerações Finais . . . . .</b>	30
<b>3 FUNDAMENTOS SOBRE CODIFICAÇÃO DE VÍDEO . . . . .</b>	31
<b>3.1 Predição . . . . .</b>	32
<b>3.2 Tratamento de Resíduo . . . . .</b>	36
<b>3.3 O Padrão HEVC . . . . .</b>	37
<b>3.3.1 Configurações de Codificação e Estruturas de Particionamento . . . . .</b>	39
<b>3.3.2 Predição Intraquadro . . . . .</b>	41
<b>3.3.3 Predição Interquadros . . . . .</b>	44
<b>3.4 Considerações Finais . . . . .</b>	50
<b>4 PROPOSTAS DA LITERATURA PARA A CODIFICAÇÃO DE VÍDEOS OMNIDIRECIONAIS . . . . .</b>	51
<b>4.1 Esforços para a Padronização . . . . .</b>	51
<b>4.2 Adaptações na Predição Intraquadro . . . . .</b>	54
<b>4.3 Adaptações na Predição Interquadros . . . . .</b>	56
<b>4.4 Adaptações para a Transmissão . . . . .</b>	58
<b>4.5 Considerações Finais . . . . .</b>	60
<b>5 AVALIAÇÃO DO COMPORTAMENTO DA CODIFICAÇÃO DE VÍDEOS OMNIDIRECIONAIS . . . . .</b>	62
<b>5.1 Metodologia de Avaliação . . . . .</b>	63
<b>5.1.1 Sequências Avaliadas e Respectivas Resoluções . . . . .</b>	63
<b>5.1.2 Avaliação de Qualidade . . . . .</b>	65
<b>5.1.3 Avaliações Baseadas em Bandas . . . . .</b>	66
<b>5.2 Avaliação das Ocorrências de <i>Skip</i> . . . . .</b>	67
<b>5.2.1 Avaliação Espacial das Ocorrências de <i>Skip</i> . . . . .</b>	67

5.2.2	Avaliação das Ocorrências de <i>Skip</i> de Acordo com a Variância . . . . .	71
<b>5.3</b>	<b>Avaliação dos Vetores de Movimento da Busca Raster . . . . .</b>	<b>74</b>
5.3.1	Avaliação da Distribuição Angular dos Vetores da Busca Raster . . . . .	76
5.3.2	Avaliação da Magnitude dos Vetores de Movimento . . . . .	79
5.3.3	Avaliação do Impacto da Redução da Área de Busca . . . . .	80
<b>5.4</b>	<b>Avaliação do Impacto da Estimação de Movimento Fracionária . . . . .</b>	<b>83</b>
<b>5.5</b>	<b>Avaliação dos Tamanhos de Bloco na Predição Intraquadro . . . . .</b>	<b>86</b>
<b>5.6</b>	<b>Considerações Finais . . . . .</b>	<b>89</b>
<b>6</b>	<b>TÉCNICAS DE REDUÇÃO DE COMPLEXIDADE PARA A CODIFICAÇÃO DE VÍDEOS OMNIDIRECIONAIS . . . . .</b>	<b>91</b>
6.1	<i>InterTech: Early Skip</i> . . . . .	92
6.2	<i>InterTech: Reduced SR</i> . . . . .	97
6.3	<i>InterTech: Reduced FME</i> . . . . .	99
6.4	<i>IntraTech: Reduced Intra Sizes</i> . . . . .	101
6.5	Técnicas <i>InterTech</i> e <i>IntraTech</i> . . . . .	106
<b>7</b>	<b>RESULTADOS EXPERIMENTAIS . . . . .</b>	<b>109</b>
7.1	Resultados de <i>Early Skip</i> . . . . .	111
7.2	Resultados de <i>Reduced SR</i> . . . . .	114
7.3	Resultados de <i>Reduced FME</i> . . . . .	116
7.4	Resultados das técnicas <i>InterTech</i> integradas . . . . .	118
7.5	Resultados de <i>Reduced Intra Sizes</i> . . . . .	122
7.6	Resultados das técnicas <i>IntraTech</i> integradas . . . . .	124
<b>8</b>	<b>CONCLUSÃO . . . . .</b>	<b>129</b>
<b>9</b>	<b>TRABALHOS SUBMETIDOS E PUBLICADOS . . . . .</b>	<b>132</b>
<b>9.1</b>	<b>Trabalhos na Área da Dissertação . . . . .</b>	<b>132</b>
9.1.1	The Impacts of Equirectangular 360-degrees Videos in the Intra-Frame Prediction of HEVC . . . . .	132
9.1.2	Spatially Adaptive Intra Mode pre-Selection for ERP 360 Video Coding (aceito para publicação) . . . . .	132
<b>9.2</b>	<b>Trabalhos Fora da Área da Dissertação . . . . .</b>	<b>132</b>
9.2.1	Speedup evaluation of HEVC parallel video coding using Tiles . . . . .	132
9.2.2	Memory-aware Workload Balancing Technique based on Decision Trees for Parallel HEVC Video Coding ( <i>submetido</i> ) . . . . .	132
9.2.3	HEVC Video Coding Using Decision Trees for a Memory-Friendly Tiles Workload Balance ( <i>Best Paper Award</i> ) . . . . .	133
<b>REFERÊNCIAS</b>	<b>134</b>	
<b>APÊNDICE A</b>	<b>VARIÂNCIAS DE CORTE DE ACORDO COM RESOLUÇÃO, NÚMERO DE BANDAS, <i>CUTOFF</i>, TAMANHO DE BLOCO E QP</b>	<b>140</b>
<b>APÊNDICE B</b>	<b>DISTRIBUIÇÃO DOS TAMANHOS DE PU DE ACORDO COM A LINHA DE CTUS . . . . .</b>	<b>170</b>
<b>ANEXO A</b>	<b>PRIMEIRO QUADRO DAS SEQUÊNCIAS OMNIDIRECIONAIS UTILIZADAS . . . . .</b>	<b>174</b>

# 1 INTRODUÇÃO

Os avanços na microeletrônica e nos processos de fabricação de dispositivos semicondutores tornaram possível a miniaturização de equipamentos eletrônicos com alto poder de processamento, além de tornar produtos eletrônicos mais acessíveis para a população em geral. Estes fatores levaram a uma popularização de vários dispositivos, tais como *smartphones*, *tablets*, computadores *notebook* e *desktop*, entre outros dispositivos que possuem várias funcionalidades além de serem capazes de capturar e reproduzir vídeos digitais. A popularização destes dispositivos, aliado também à popularização do acesso à internet, levou a um aumento expressivo no consumo de vídeos digitais. Dados da CISCO indicam que até 2022, 82% do tráfego de dados da internet será composto por vídeos digitais (CISCO, 2019a), enquanto que em dispositivos móveis vídeos digitais vão corresponder a 79% do tráfego de dados (CISCO, 2019b). Os contínuos avanços nas tecnologias de captura e exibição de vídeo têm levado à significativas melhoria na qualidade dos vídeos digitais, e várias destas melhorias são atingidas devido as novas tecnologias para aumentar a imersão ou interatividade dos usuários com os vídeos.

Dentre os tipos de imersão mais comuns, podemos citar vídeos tridimensionais (3D), vídeos em *light-field*, e vídeos omnidirecionais. Vídeos 3D consistem em vídeos cuja captura é realizada com múltiplas câmeras, ligeiramente deslocadasumas das outras. Com estes vídeos e um sistema de reprodução adequado, o usuário pode ter a sensação de profundidade enquanto assiste aos vídeos, e em alguns casos, também é possível assistir a diferentes pontos de vista da cena. No caso de vídeos *light-field*, a captura é realizada utilizando câmeras especiais, capazes de capturar tanto a intensidade quanto a direção em que a luz incide sobre a lente. Essa técnica possibilita, entre outras coisas, que o foco da imagem seja ajustado após a captura. Por fim, vídeos omnidirecionais são capturados utilizando câmeras com múltiplas lentes de grande ângulo, de forma que é possível interpolar as vistas das várias lentes para criar uma representação esférica da cena ao redor da câmera. Com esta representação, o usuário é capaz de assistir a um ponto de vista arbitrário desta esfera, como se estivesse imerso na cena.

Vídeos 3D já vêm sendo estudados há muitos anos, e embora muitos avanços tenham sido atingidos, como televisores domésticos 3D e *videogames* portáteis com telas 3D (FRANKLIN, 2010), essa tecnologia não se consolidou comercialmente e atualmente este tipo de conteúdo está recebendo menos atenção da indústria e da academia. A tecnologia de *light-fields* também já vem sendo estudada há algum tempo, e embora câmeras tenham sido lançadas para atender ao mercado consumidor, esta tecnologia ainda está dando os primeiros passos para se estabelecer no mercado. Assim como os anteriores, vídeos omnidirecionais também não são uma ideia nova, mas com a recente popularização de óculos de realidade virtual (*Virtual Reality - VR*) como o Oculus Rift (OCULUS, 2019) e HTC Vive (VIVE, 2019), além de dispositivos que associados a um *smartphone* são capazes de prover VR (SAMSUNG, 2019a), aplicações com este tipo de mídia têm recebido muita atenção da indústria e da academia.

Quando vídeos digitais estão em seu formato original, estes demandam uma quantidade excessivamente grande de dados para serem representados, de forma que sua manipulação comercial (transmissão e armazenamento) se torna infactível. Além disso, vídeos convencionais e omnidirecionais costumam ter resoluções diferentes. No que tange a vídeos convencionais, a resolução  $1920 \times 1080$  costuma ser o suficiente para a maioria das aplicações, e para aplicações que demandem uma qualidade ainda superior, a resolução 4k representa uma melhora significativa e suficiente para aplicações comerciais. Entretanto, quando vídeos omnidirecionais são considerados, as resoluções 8k empregadas atualmente ainda deixam a desejar mesmo para aplicações convencionais (VISBIT, 2016). Isso ocorre porque vídeos omnidirecionais representam um campo de visão significativamente maior do que vídeos convencionais e aplicações em VR requerem que o *display* esteja muito próximo aos olhos do usuário – o que também demanda um aumento na resolução.

Da necessidade de reduzir o volume de dados necessário para a representação de vídeos digitais surgiram os padrões de codificação de vídeo: conjuntos de especificações capazes de gerar técnicas que comprimem vídeos digitais e que, com isso, tornam os mesmos mais adequados para distribuição.

Estes padrões de codificação de vídeo exploram diversas características dos vídeos, como redundâncias espaciais e temporais, para atingir altas taxas de compressão. Além disso, cada geração de padrões de codificação incorpora novas ferramentas ou melhora ferramentas já existentes, com o objetivo de se adequar à evolução dos vídeos e obter maiores taxas de compressão.

A codificação de vídeos convencionais é um problema conhecido por apresentar uma grande complexidade computacional e múltiplos trabalhos foram desenvolvidos para reduzir tal complexidade. A partir disso, e sabendo que vídeos omnidirecionais apresentam resoluções significativamente maiores, percebe-se que o problema

da alta complexidade computacional já presente na codificação de vídeos convencionais é agravado pela maior quantidade de dados presente em vídeos omnidirecionais. Sendo assim, fica claro que a redução de complexidade de vídeos omnidirecionais é uma área de pesquisa de suma importância para uma maior popularização de vídeos omnidirecionais.

Devido à falta de um padrão de codificação específico para a codificação de vídeos omnidirecionais, estes são codificados com os mesmos padrões e ferramentas utilizados na codificação de vídeos convencionais. Entretanto, para que isso seja possível se fazem necessárias uma série de etapas de pré- e pós-processamento, como será discutido no Capítulo 2. Dentre estas etapas, uma das mais cruciais é a projeção: uma técnica que permite representar vídeos omnidirecionais (uma superfície esférica) de forma retangular, assim como vídeos convencionais.

Embora existam múltiplas maneiras de realizar a projeção, a projeção equirretangular (ERP) é a mais comumente utilizada, e desta forma, esta é a projeção empregada neste trabalho. Ainda que a projeção ERP seja capaz de tornar a codificação de vídeos omnidirecionais possível, ela causa uma série de distorções no vídeo que não são presentes em vídeos convencionais, e isso pode interferir no comportamento das ferramentas de codificação.

Sendo assim, a hipótese geral deste trabalho é que o comportamento das ferramentas de codificação de vídeo apresentam um comportamento diferente durante a codificação de vídeos omnidirecionais na projeção ERP quando comparado à codificação de vídeos convencionais devido às distorções causadas pela projeção ERP.

Neste contexto, o objetivo deste trabalho é explorar as distorções geradas pela projeção ERP para reduzir a complexidade do processo de codificação de vídeos omnidirecionais. Para atingir tal objetivo, o trabalho apresenta uma série de avaliações sobre os efeitos das distorções geradas pela projeção ERP no processo de codificação. Os resultados das avaliações são utilizados como base para o desenvolvimento de técnicas de redução de complexidade que atuam em diferentes etapas do processo de codificação de vídeo. Os resultados mostram que é possível reduzir a complexidade da codificação explorando as distorções geradas pela projeção ERP nos vídeos omnidirecionais sem prejudicar a eficiência de compressão.

## 1.1 Organização

O restante deste trabalho está dividido da seguinte forma. O Capítulo 2 discorre sobre as características e etapas de processamento de vídeo omnidirecionais. O Capítulo 3 fala sobre as principais ferramentas de codificação de vídeo atuais e suas respectivas implementações no padrão *High Efficiency Video Coding* (HEVC). O Capítulo 4 expõe os principais esforços sendo feitos em direção à padronização da distribuição

e codificação de vídeos omnidirecionais, além de discutir trabalhos propostos para melhorar os padrões de codificação de vídeo já existentes quando aplicados a vídeos omnidirecionais. O Capítulo 5 apresenta um conjunto de avaliações realizadas sobre a codificação de vídeos omnidirecionais, onde as características específicas destes são identificadas e discutidas. Em seguida, o Capítulo 6 apresenta as técnicas de redução de complexidade desenvolvidas a partir das avaliações realizadas. A seguir, o Capítulo 7 discorre sobre os resultados obtidos ao utilizar-se as técnicas de redução de complexidade desenvolvidas durante a codificação de vídeos omnidirecionais, além de comparar os resultados obtidos com outros trabalhos da literatura. O Capítulo 8 apresenta algumas considerações e conclusões sobre o que foi apresentado no restante do trabalho. Por fim, o Capítulo 9 relata as publicações geradas durante o desenvolvimento deste trabalho.

## 2 VÍDEOS OMNIDIRECIONAIS, SUAS CARACTERÍSTICAS E APLICAÇÕES

Primeiramente, este capítulo busca trazer as características de vídeos digitais em geral, e os métodos utilizados para representá-los. Em seguida, são mostradas as características específicas de vídeos omnidirecionais, além de algumas de suas aplicações. Por fim, é apresentado o fluxo de processamento de vídeos omnidirecionais, desde a captura até a execução.

### 2.1 Características de Vídeos Digitais em Geral

Um vídeo digital pode ser definido como uma sequência de imagens ou quadros (em inglês, *frames*), que é apresentada ao espectador com uma determinada frequência de atualização para que seja percebida sensação de movimento. Essa frequência de atualização é denominada taxa de quadros por segundo, e para que o espectador tenha uma sensação de movimento fluído é necessário que esta taxa seja de pelo menos 24 ou 30 quadros por segundo (GONZALEZ; WOODS, 2006).

Além disso, a representação de imagens e vídeos coloridos está associada ao funcionamento do sistema visual humano. Dentre as células que compõem o olho humano, vale ressaltar as células fotossensíveis bastonetes e cones. Os bastonetes são muito numerosos e muito sensíveis à luminosidade, sendo assim são capazes de distinguir formas mesmo com pouca luminosidade. Entretanto, os bastonetes geram imagens com baixo nível de detalhes. Já os cones são menos numerosos e capazes de distinguir as cores, gerando imagens mais ricas em detalhes. No entanto, os cones precisam de maior intensidade luminosa para que sejam excitados (POLLACK, 2006).

Dada a constituição do sistema visual humano, a representação de imagens e vídeos coloridos se baseia em decompor a imagem em múltiplos canais que excitam diferentes partes do sistema visual humano de formas distintas. A forma como a imagem é decomposta em canais é denominada espaço de cores, sendo que os espaços RGB e YCbCr estão dentre os mais comumente utilizados (SHI; SUN, 2008).

O espaço de cores RGB recebe este nome por decompor o sinal em três canais

distintos, representando as cores primárias de acordo com a teoria aditiva de cores: vermelho (**Red**), verde (**Green**) e azul (**Blue**). Sendo assim, uma dada imagem ou quadro é decomposto em três matrizes, cada uma contendo as informações de um canal de cor primária. Já o espaço de cores YCbCr decompõe o sinal em canais de luminância (Y), crominância azul (Cb) e crominância vermelha (Cr). O canal de luminância é responsável por representar a intensidade luminosa (ou seja, a imagem em preto e branco), enquanto que as crominâncias azul e vermelha são responsáveis por representar as cores da imagem.

Embora o espaço de cores RGB seja mais comum nos dispositivos de visualização, esta representação não costuma ser utilizada no processo de compressão de imagem e vídeo. Isso se dá porque os canais R, G e B do espaço RGB possuem uma alta correlação, e isso dificulta o processamento das informações de cor de forma independente (RICHARDSON, 2002). Com a representação no espaço YCbCr, as informações de luminosidade e cor podem ser processadas de maneira independente, potencializando a eficiência do processo de codificação de vídeos digitais.

O processo de codificação de vídeo utiliza diversas ferramentas que exploram características presentes na representação dos vídeos digitais. Dentre as principais características está a redundância de dados. Esta redundância pode ser observada em três domínios principais: espacial, temporal, e entrópico.

A redundância espacial se dá pela redundância de informação dentro de um mesmo quadro, ou seja, é uma redundância intra-quadro (GHANBARI, 2003). Esta redundância se manifesta tanto no domínio espacial quanto no domínio das frequências. É comum que várias regiões de uma imagem ou quadro tenham pixels com valores muito próximos ou iguais, sendo assim, a redundância espacial está ligada à correlação entre os valores destes pixels. Caso esta imagem ou quadro seja dividido em blocos e uma transformada seja aplicada a cada bloco individualmente, a maior parte da energia estará concentrada nos coeficientes de baixa frequência, manifestando-se assim a redundância espacial no domínio das frequências.

A redundância temporal, por outro lado, está ligada à correlação entre quadros temporalmente vizinhos e também pode ser chamada de redundância inter-quadros (GHANBARI, 2003). Como é recomendado que a taxa de exibição – e consequentemente, de captura – de um vídeo seja de ao menos 24 quadros por segundo (GONZALEZ; WOODS, 2006), devem existir ao menos 24 quadros para representar um segundo de vídeo, o que faz com que haja muita redundância entre quadros temporalmente sucessivos. Em alguns casos essa redundância se dá de forma direta, visto que muitos pixels mantêm seus valores (ou são alterados ligeiramente) de um quadro para o próximo, enquanto que em outros casos essa redundância se dá de forma indireta, visto que um conjunto de pixels se desloca de um quadro para o próximo, representando o movimento da cena.

Por fim, existe a redundância entrópica. Essa redundância está relacionada com a probabilidade de ocorrência dos símbolos codificados e mede a quantidade média de informação transmitida por símbolo (SHI; SUN, 2008). A redundância entrópica está presente em todo o tipo de mídia: texto, imagem, áudio, vídeo, etc. No caso de vídeos, a redundância entrópica é observada tanto antes quanto depois da codificação. Antes da codificação, a redundância é visível nos símbolos utilizados para representar as cores dos diferentes pixels ao longo do vídeo, tanto espacial quanto temporalmente. Já a redundância após a codificação se manifesta nas decisões tomadas pelo codificador, como a distribuição do vídeo em blocos, o tipo e modo de codificação utilizados, os coeficientes das transformadas empregadas, entre outros. Mais detalhes sobre a redundância entrópica e como ela é explorada são apresentados na Seção 3.2.

## 2.2 Características e Aplicações de Vídeos Omnidirecionais

Vídeo omnidirecional, também conhecido como vídeo esférico ou 360°, é um tipo de conteúdo criado para propiciar experiências imersivas aos usuários. Diferente de vídeos convencionais, vídeos omnidirecionais cobrem todo o campo visual de  $360^\circ \times 180^\circ$  e representam a superfície interna de uma esfera ao redor do usuário. Esse tipo de conteúdo propicia uma sensação de "estar lá", dentro da cena, visto que o usuário pode olhar ao redor livremente e assistir ao que julgar mais interessante, assim como humanos na vida real. Um exemplo de imagem omnidirecional em formato esférico é apresentado na Figura 1, que representa a visão frontal e traseira do primeiro quadro do vídeo *PoleVault*, retirado das Condições Comuns de Teste para Vídeos Omnidirecionais (CTCs360) (BOYCE et al., 2017).

Quando associados a um *display* montado na cabeça (*head-mounted display* - HMD), como um óculos de realidade virtual, vídeos omnidirecionais têm a capacidade de propiciar uma imersão nunca antes vista, e essa imersão tem encontrado aplicações nas mais diversas áreas, como entretenimento, negócios imobiliários, edu-



Figura 1 – Vista frontal e traseira do primeiro quadro do vídeo *PoleVault*

cação/treinamento e saúde.

No campo do entretenimento, destacam-se vídeos em parques de diversões, passeios em lugares paradisíacos e da prática de esportes radicais. Já nos negócios imobiliários, o uso de vídeos omnidirecionais tem facilitado a visita dos imóveis. Tipicamente, um cliente costuma visitar uma série de imóveis até decidir qual é a melhor opção e isso é um processo que consume muito tempo e, possivelmente, dinheiro. Com a popularização de vídeos omnidirecionais e dispositivos de VR, empresas do setor imobiliário têm feito vídeos omnidirecionais do interior dos imóveis para que os possíveis clientes possam visitar múltiplas propriedades sem a necessidade de se deslocar fisicamente até elas, acelerando o processo de compra/aluguel e reduzindo custos (GLEB, 2018).

Na área da educação e treinamento, vídeos omnidirecionais podem ser usados para reforçar aulas com viagens de campo, demonstrar procedimentos médicos em detalhes sem interferir no trabalho dos profissionais, ou até mesmo apresentar o ambiente de trabalho a novos funcionários antes mesmo que estes cheguem aos seus postos (ROZAKIS, 2019). Já num misto de entretenimento e saúde, o uso de VR e vídeos omnidirecionais se mostrou muito eficaz na redução de dor e ansiedade durante procedimentos hospitalares em pacientes pediátricos (DETREMPE, 2017).

Embora vídeos omnidirecionais tenham ganhado popularidade nos últimos anos, uma maior popularização deste tipo de conteúdo ainda apresenta desafios. Um dos desafios reside na largura de banda necessária para a reprodução destes vídeos em plataformas de *streaming*. Como vídeos omnidirecionais representam todo o arredor do usuário e costumam ser reproduzidos em *displays* muito próximos dos olhos, estes vídeos possuem uma resolução consideravelmente maior do que vídeos convencionais. Mesmo possuindo resoluções maiores do que vídeos convencionais, muitos usuários reclamam que os vídeos omnidirecionais atuais apresentam baixas resoluções, e discute-se que os *displays* presentes nos HMDs precisam ser melhorados para que a experiência imersiva se torne mais natural (VISBIT, 2016).

Como vídeos omnidirecionais representam toda a cena ao redor do usuário e possuem características interativas, o fluxo de processamento e distribuição possui algumas particularidades quando comparada a de vídeos convencionais. A Seção a seguir busca trazer uma visão geral do fluxo de processamento de vídeos omnidirecionais.

## 2.3 Fluxo de Processamento de Vídeos Omnidirecionais

Esta Seção tem o objetivo de apresentar as principais etapas do fluxo de processamento de vídeos omnidirecionais. A Figura 2 demonstra um modelo deste fluxo de processamento, onde inicialmente o vídeo é capturado utilizando-se uma câmera especial com múltipla lentes, e em seguida passa por um processo de costura para



Figura 2 – Fluxo de processamento de vídeos omnidirecionais

criar uma representação esférica da cena a partir do resultado das diversas lentes. Em seguida este vídeo esférico passa por uma projeção para dar origem a uma representação planar da cena. A versão planar do vídeo omnidirecional é então codificada, armazenada/transmitida e, por fim, decodificada pelo usuário. Uma vez decodificado este vídeo é mapeado novamente numa superfície esférica através da projeção inversa e um determinado ponto de vista (*viewport*) é extraído desta de acordo com alguns parâmetros relativos ao usuário e ao *software* de reprodução. Por fim, este *viewport* é reproduzido pelo usuário através de algum tipo de *display*. As Seções a seguir esclarecem de forma mais detalhada cada uma das etapas.

### 2.3.1 Captura e Costura

Vídeos omnidirecionais representam uma cena em todas as direções, como uma casca de esfera cuja câmera ou o espectador estão localizados no centro desta esfera. Dada esta característica deste tipo de conteúdo, a captura de vídeos omnidirecionais não pode ser realizada com uma única câmera convencional. Desta forma, a captura de vídeos omnidirecionais requer o uso de câmeras com ao menos duas lentes de grande ângulo, de forma que qualquer porção da cena esteja presente no campo de captura de ao menos uma das lentes. As duas principais maneiras de atingir esta restrição são: (1) realizar a captura utilizando-se múltiplas câmeras convencionais, ou (2) utilizar uma câmera específica para a captura de vídeos omnidirecionais.

Quando o uso de diversas câmeras é considerado, empresas como GoPro vendem conjuntos de câmeras equipadas com uma armação capaz de manter as câmeras posicionadas de forma adequada, além de sincronizar a captura destas. Um exemplo destes conjuntos é o GoPro Omni (GOPRO, 2019a), que consiste de uma armação cúbica onde seis câmeras GoPro HERO4 Black (GOPRO, 2019b) são acopladas. Através desta armação e com um *firmware* específico para o GoPro Omni, todas as câmeras atuam de forma sincronizada, isto é, mantém as mesmas configu-



Figura 3 – Exemplos de câmeras omnidirecionais. (a) GoPro Omni, (b) Samsung Gear 360, (c) LG 360 CAM

rações e iniciam/terminam sua captura simultaneamente. O conjunto GoPro Omni é apresentado na Figura 3 (a).

Embora seja possível realizar a captura com diversas câmeras, na maioria dos casos a captura é feita com câmeras especialmente projetadas para a captura de vídeos omnidirecionais. Exemplos destas câmeras são a Samsung Gear 360 (SAMSUNG, 2019b) e a LG 360 CAM (LG, 2019). Em ambos os casos, as câmeras são compostas por duas lentes de grande ângulo posicionadas em lados opostos do equipamento, de forma que capturam a cena em todas as direções. Como todas as lentes, conexões e botões fazem parte da estrutura da câmera, estas são mais compactas do que as opções que utilizam várias câmeras acopladas numa armação. As câmeras Samsung Gear 360 e LG 360 CAM são apresentadas na Figura 3 (b) e (c), respectivamente.

Como são utilizadas múltiplas lentes para realizar a captura, o resultado desta é um conjunto com múltiplos vídeos planares representando diferentes pontos de vista. Para que este conjunto de vídeos planares se torne um vídeo esférico, estes precisam passar por um processo chamado costura. O primeiro passo da costura consiste em encontrar correspondências entre os vídeos que permitam o alinhamento destes, e com isso minimize o erro quando os vídeos planares são sobrepostos (Szeliski et al., 2007). Em seguida, é feita uma série de ajustes para corrigir os efeitos causados pelo uso de diversas câmeras, como ajustar o brilho dos vídeos, reduzir distorções cromáticas, corrigir diferenças causadas por diferenças na exposição dos diferentes vídeos, entre outros. Uma vez que os vídeos tenham sido alinhados e os efeitos das múltiplas lentes tenham sido corrigidos, é possível mapear estes vídeos para o interior de uma superfície esférica.

De acordo com as especificações da câmera, a costura pode ser realizada no próprio dispositivo ou pode demandar o uso de um *software* externo num computador. Câmeras como a Pilot Era (PISOFTECH, 2019) realizam a costura em tempo real e entregam o vídeo esférico pronto para o usuário, enquanto que câmeras como a

GoPro Omni (GOPRO, 2019a) entregam vários vídeos planares que precisam ser costurados posteriormente pelo usuário.

### **2.3.2 Projeção Equirretangular - ERP**

Após a costura o conjunto de vídeos planares é transformado em um vídeo esférico, pronto para proporcionar experiências imersivas. Entretanto, como estes vídeos possuem uma resolução consideravelmente maior do que vídeos convencionais, estes também demandam uma quantidade de dados significativamente maior para serem representados. Além disso, como apresentado no Capítulo 1, vídeos digitais precisam ser comprimidos para que possam ser manipulados de forma prática, o que traz um novo problema, pois todos os padrões de codificação atuais foram desenvolvidos para lidar com vídeos retangulares planares, e não esféricos. Sendo assim, um vídeo omnidirecional já costurado deve ser mapeado para uma superfície planar novamente para que possa ser processado pelos padrões de codificação atuais.

Projetar um vídeo omnidirecional numa superfície planar retoma um problema antigo da civilização: criar mapas cartográficos da superfície terrestre. Uma das propriedades de projeções cartográficas – e consequentemente, projeções de vídeos omnidirecionais – é que a superfície da esfera sempre será distorcida de alguma forma: a forma dos objetos é alterada, o tamanho dos objetos é modificado, a posição relativa entre objetos é alterada, entre outros. Estas distorções podem ser prejudiciais à etapa de codificação, visto que os padrões de codificação de vídeo atuais foram desenvolvidos para lidar com as redundâncias presentes em vídeos convencionais e tais distorções podem alterar estas redundâncias. Mais detalhes sobre o funcionamento das ferramentas de codificação de vídeo serão apresentados no Capítulo 3. Visando melhorar o desempenho dos codificadores de vídeo atuais quando aplicados a vídeos omnidirecionais, diversas projeções foram propostas na literatura.

A projeção equirretangular (ERP) é a mais encontrada nos trabalhos sobre vídeos omnidirecionais na literatura e, como mencionado no capítulo anterior, será a projeção foco deste trabalho. Isso se deve principalmente a facilidade de conversão entre a representação esférica e planar, e por esta projeção representar de forma intuitiva uma superfície esférica num espaço retangular. De forma simplista, é possível dizer que a projeção ERP transforma cada paralelo da esfera em uma linha do retângulo (YE; ALSHINA; BOYCE, 2017), de modo que não existe distorção vertical na imagem. Entretanto, os paralelos da esfera possuem um raio diferente de acordo com sua localização: o equador possui o maior raio entre todos os paralelos e paralelos mais próximos aos polos apresentam raios cada vez menores. Dessa forma, o paralelo central (linha do equador) é mapeado perfeitamente para o centro do retângulo, enquanto que os demais paralelos precisam ser esticados em diferentes escalas para que possam preencher a largura do retângulo. No caso mais extremo, o polo norte e



Figura 4 – Primeiro quadro do vídeo *PoleVault* na projeção ERP

polo sul, que constituem um único ponto no espaço, precisam ser esticados demasia-damente até a largura do retângulo. Isso faz com que a região central do quadro ERP seja representado de forma fiel, enquanto que as regiões próximas aos polos sejam representadas de forma distorcida. O primeiro quadro do vídeo *PoleVault* na projeção ERP é apresentado na Figura 4.

Dentre as outras projeções disponíveis, é possível citar a projeção parabólica de Craster, que é semelhante à projeção ERP caso não fosse realizado o esticamento horizontal (YE; ALSHINA; BOYCE, 2017). Existem também as projeções *cubemap* e octaédrica, que consistem em projetar a superfície da esfera nas faces internas de um cubo e um octaedro, respectivamente, que posteriormente são desmontados e as faces rearranjadas (YE; ALSHINA; BOYCE, 2017). Já a projeção em esfera segmentada (*segmented sphere projection*) converte conjuntos de paralelos vizinhos em faixas de um vídeo retangular, enquanto que as regiões polares são convertidas em círculos preenchidos ao redor para dar origem a quadrados (YE; ALSHINA; BOYCE, 2017). Além das projeções citadas anteriormente, muitas outras projeções podem ser utilizadas para codificar vídeos omnidirecionais e a criação de novas projeções consiste em uma área de pesquisa por si só. Entretanto, cada projeção utiliza métodos diferentes para mapear o vídeo esférico em um vídeo planar retangular e, a partir disso, cada projeção gera distorções diferentes no vídeo projetado.

### 2.3.3 Codificação, transmissão/armazenamento e decodificação

Devido à falta de um padrão específico para a codificação de vídeos omnidirecionais atualmente, estes são codificados da mesma forma que vídeos convencionais, isto é, com os mesmos padrões e ferramentas de codificação. Esta abordagem tende a não resultar na melhor eficiência de codificação, visto que as projeções utilizadas para permitir a codificação causam uma série de distorções no vídeo que podem in-

terferir no comportamento e eficiência das ferramentas de codificação convencionais – que não foram concebidas para lidar com este tipo de conteúdo.

Existem esforços das entidades padronizadoras para desenvolver ferramentas específicas e padronizar a codificação de vídeos omnidirecionais, como será tratado na Seção 4.1, entretanto, estes esforços tendem a surtir efeito apenas nos padrões de codificação que serão lançados no futuro e não resolvem o problema da distribuição de vídeos omnidirecionais no cenário corrente.

Sendo assim, vídeos omnidirecionais são codificados por padrões de codificação de vídeo convencionais, assim como será abordado no Capítulo 3. Em seguida estes vídeos codificados são distribuídos de forma convencional, até chegarem ao usuário final. Do lado do usuário, o vídeo é decodificado da mesma forma que um vídeo convencional e em seguida é reproduzido.

### 2.3.4 Renderização, Seleção de *Viewport* e Reprodução

Uma vez que o vídeo é recebido e decodificado pelo usuário, este tem um vídeo omnidirecional projetado. Entretanto, antes da reprodução este vídeo precisa ser renderizado para a superfície esférica novamente através da projeção inversa.

Visto que o campo de vista do sistema visual humano não cobre toda a superfície da esfera, é necessário que um campo de vista (*viewport*) seja extraído da esfera para ser exibido ao usuário. A seleção do *viewport* depende de um fator externo ao vídeo que indique qual região da esfera deve ser extraída e este fator costuma ser o conjunto de sensores do HMD ou um dispositivo como um mouse ou *joystick*.

Considerando o caso em que o usuário está usando um HMD e o diagrama apresentado na Figura 5, a extração do *viewport* pode ocorrer da seguinte forma. Dado que o vídeo omnidirecional está representado pela esfera e o usuário está localizado na origem  $O$ , o conjunto de sensores do HMD deve gerar um vetor que caracterize a posição da cabeça do usuário e incida no ponto  $O'$  da esfera. A partir deste ponto, é gerado um plano  $ABCD$  tangente à esfera no ponto  $O'$ , que caracteriza o *viewport*. Por fim, a superfície da esfera é projetada sobre o plano  $ABCD$  utilizando uma projeção retilínea (YU; LAKSHMAN; GIROD, 2015). Conforme o usuário assiste ao vídeo,

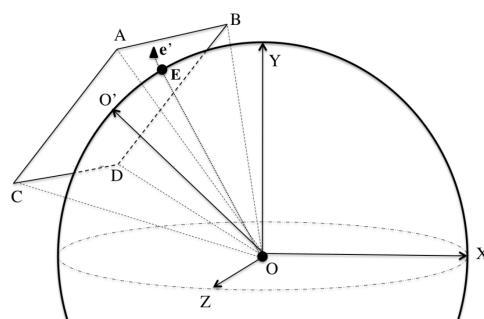


Figura 5 – Exemplo de extração de *viewport* (YU; LAKSHMAN; GIROD, 2015)

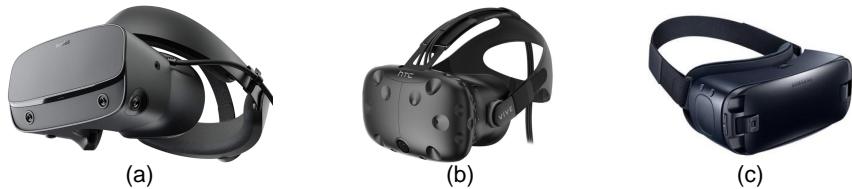


Figura 6 – Exemplos de HMDs. (a) Oculus Rift, (b) HTC Vive, (c) Samsung Gear VR

os sensores do HMD se atualizam e o ponto  $O'$  é atualizado, atualizando o *viewport* exibido ao usuário.

Uma vez que um *viewport* foi extraído do vídeo omnidirecional, este pode ser exibido ao usuário através de diferentes tipos de *displays*. O uso de HMDs propicia a maior imersão, visto que estes dispositivos foram desenvolvidos justamente com este propósito. Entre os HMDs mais utilizados, podemos citar o Oculus Rift (OCULUS, 2019) e o HTC Vive (VIVE, 2019), apresentados na Figura 6 (a) e (b), respectivamente, além do Samsung Gear VR (SAMSUNG, 2019b), apresentado na Figura 6 (c), que possibilita inserir um *smartphone* e utilizá-lo como HMD. Além destes dispositivos, é possível assistir vídeos omnidirecionais em *displays* convencionais, caso em que o *viewport* é determinado a partir de outros periféricos.

## 2.4 Considerações Finais

Este capítulo apresentou conceitos referentes a vídeos digitais em geral, como sua representação e como exploram o sistema visual humano. Em seguida, foram mostradas as características específicas dos vídeos omnidirecionais em relação a vídeos convencionais, além de apresentar o pré- e pós-processamento necessário para que vídeos omnidirecionais possam ser manipulados por ferramentas e serviços desenvolvidos para a manipulação de vídeos convencionais.

Uma das etapas de pré-processamento que merece destaque é a projeção, que permite representar um vídeo omnidirecional de natureza esférica na forma de um retângulo planar. A projeção permite que vídeos omnidirecionais sejam codificados através de padrões de codificação de vídeo convencionais, entretanto, a projeção insere distorções na textura do vídeo que não são presentes em vídeos convencionais, e desta forma, é possível que as ferramentas de codificação convencionais – desenvolvidas para explorar características de vídeos convencionais – não sejam capazes de entregar os melhores resultados possíveis.

Sendo assim, o próximo capítulo discute os aspectos fundamentais sobre codificação de vídeo, além de detalhar o funcionamento do padrão HEVC, para que seja possível discutir como as distorções da projeção podem interferir no funcionamento das ferramentas de codificação.

### 3 FUNDAMENTOS SOBRE CODIFICAÇÃO DE VÍDEO

Como mencionado no Capítulo 1, vídeos digitais em seu formato original demandam uma quantidade excessivamente grande de dados para serem representados, e isso impossibilita a manipulação desse tipo de mídia de forma prática. Para possibilitar a manipulação de vídeos digitais de forma prática, estes precisam ser comprimidos através de padrões de codificação de vídeo.

Padrões de codificação de vídeo consistem em normas que geram *softwares* capazes de reduzir o volume de dados necessário para representar vídeos digitais. Estes padrões de codificação buscam explorar as diversas redundâncias dentro dos vídeos para reduzir de forma significativa o volume de dados causando o menor impacto possível na qualidade dos vídeos.

Embora diferentes padrões de codificação de vídeo apresentem diferentes características, a Figura 7 apresenta o diagrama de blocos de um codificador de vídeo genérico que modela a maioria dos codificadores atuais. Segundo o diagrama da Figura 7, um quadro do vídeo a ser codificado é subdividido em blocos (esta etapa não é mostrada na Figura 7) e em seguida este bloco passa pela predição interquadros e/ou predição intraquadro. Quando a predição interquadros é utilizada é necessário que haja um conjunto de quadros de referência, como será discutido na Seção 3.1. Uma vez realizada a predição, o bloco predito é subtraído do bloco original para dar origem ao erro de predição, e este erro de predição passa por uma etapa de transformada e quantização (bloco T/Q). Por fim, os coeficientes quantizados do resíduo de predição passam pela codificação de entropia para dar origem ao *bitstream* que representa o vídeo codificado. Além disso, após quantizar os coeficientes do resíduo de predição estes passam pela quantização e transformada inversa (bloco  $T^{-1}/Q^{-1}$ ) e em seguida são somados ao bloco predito para originar o bloco reconstruído – este bloco reconstruído é idêntico aquele reconstruído pelo decodificador. Após gerar o bloco reconstruído, este passa por um conjunto de filtros e é armazenado junto com os demais blocos reconstruídos para gerar os quadros de referência armazenados na memória externa.

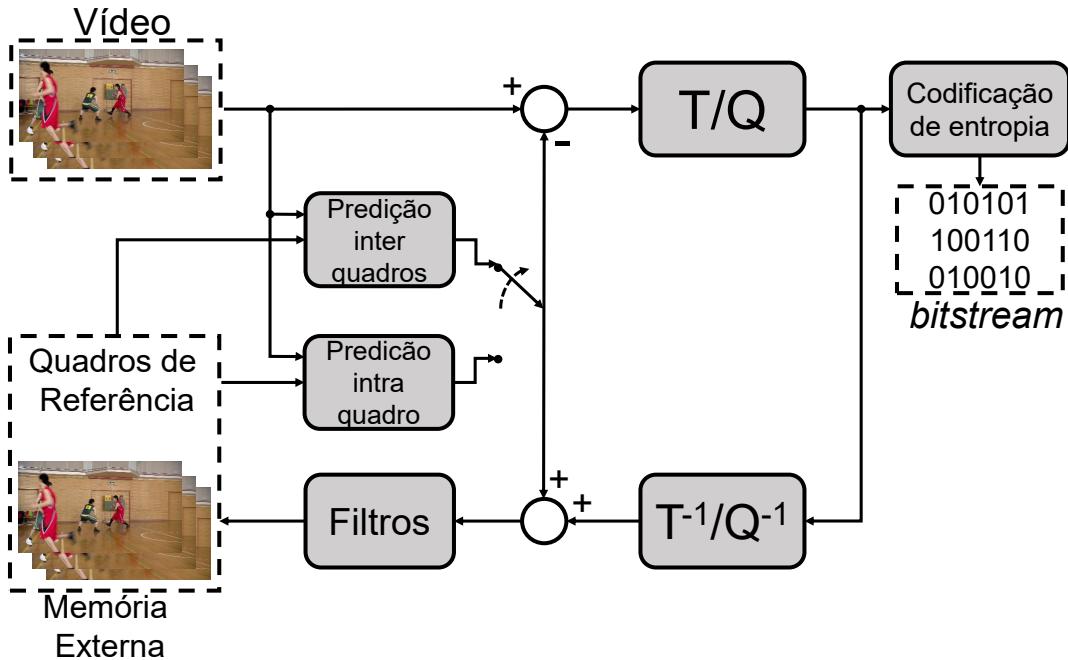


Figura 7 – Diagrama de blocos de um codificador de vídeo genérico (CORRÊA, 2018)

As próximas Seções buscam trazer os fundamentos básicos sobre as principais ferramentas dos padrões de codificação de vídeo atuais, além de trazer detalhes sobre a implementação destas ferramentas no padrão *High Efficiency Video Coding* (HEVC), o padrão utilizado neste trabalho para avaliar as técnicas propostas.

### 3.1 Predição

Como dito na Seção 2.1, vídeos digitais apresentam redundâncias espaciais e temporais e os padrões de codificação de vídeo exploram tais redundâncias para atingir altas taxas de compressão causando o menor impacto possível na qualidade dos vídeos. De modo geral, os padrões de codificação utilizam preditores para representar um bloco de amostras com base em outro bloco já codificado, isto é, representam algumas regiões do vídeo com base em outras regiões.

Para explorar a redundância espacial (dentro do mesmo quadro) os padrões de codificação de vídeo utilizam a predição intraquadro, que busca representar o bloco atual (a ser predito) com base nos blocos vizinhos já preditos. Dessa forma, é visível que existe uma dependência entre os blocos de um quadro, logo, existe uma ordem inerente à codificação dos blocos dentro de um quadro do vídeo.

Os padrões de codificação de vídeo atuais costumam implementar a ordem *raster* para codificar os blocos dentro de um quadro, isto é, os blocos são codificados da esquerda para a direita, de cima para baixo. A Figura 8 apresenta o primeiro quadro do vídeo *PoleVault*, onde estão sobrepostos  $10 \times 5$  blocos e os primeiros são numerados em ordem *raster*. Considerando que a codificação ocorre na ordem *raster* apresenta-

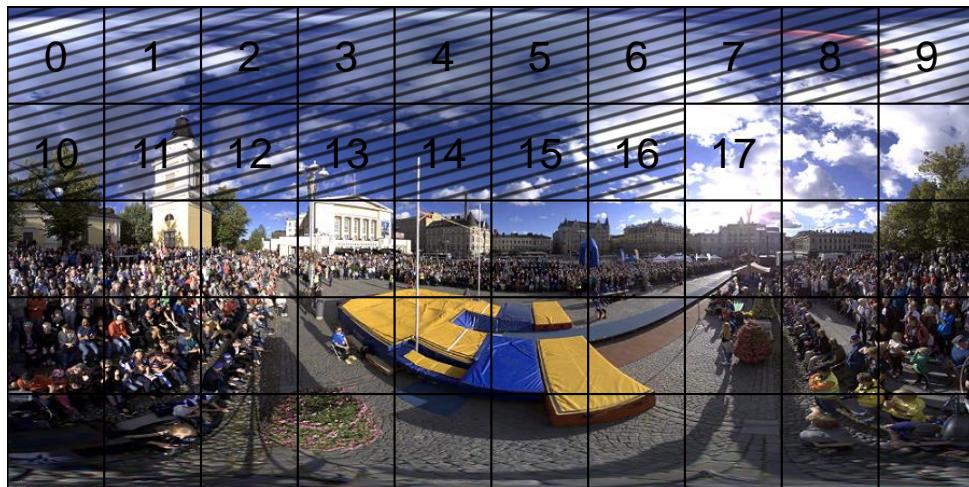


Figura 8 – Blocos em ordem raster sobre o primeiro quadro do vídeo *PoleVault*

tada, fica visível que durante a predição de um determinado bloco, todos os blocos na mesma linha à esquerda e em todas as linhas acima da atual já terão sido codificados.

Como exemplo, durante a predição do bloco número 17, todos os blocos de 0 a 16 já terão sido preditos, como mostrado pelos blocos hachurados da Figura 8. Desta forma, os blocos de 0 a 16 poderão ser usados como referência para realizar a predição do bloco 17. Fica evidente que este modelo pode apresentar problemas para realizar a predição do primeiro bloco de cada quadro, visto que não existem blocos já preditos. Nesse caso, os padrões de codificação utilizam modos de predição intraquadro especiais que não dependem de vizinhos codificados, ou extrapolam as amostras do bloco atual para formar blocos vizinhos fictícios.

Os modos de predição intraquadro costumam ser divididos em dois grupos independente do padrão de codificação: modos de predição direcionais e não-direcionais. Os modos direcionais tem o propósito de explorar texturas bem definidas dentro do bloco e baseiam-se em blocos vizinhos já codificados, enquanto que os modos não-direcionais apresentam características especiais e são diferentes de acordo com o padrão de codificação sendo empregado.

A Figura 9 apresenta um exemplo de funcionamento da predição intraquadro. Nesse caso, a região acinzentada corresponde ao bloco que deve ser predito, enquanto que as linhas tracejadas dentro deste bloco demarcam as amostras dentro do bloco. Para este exemplo, será demonstrado o funcionamento de um modo de predição direcional horizontal, ou seja, um modo que representa as amostras do bloco atual com base nas amostras do(s) bloco(s) à esquerda do bloco a ser predito.

A maneira mais simples de realizar esta predição é considerar apenas as amostras de referência mais próximas ao bloco atual (amostras  $A_0$ ,  $B_0$ ,  $C_0$  e  $D_0$ ) e replicá-las diretamente sobre as amostras do bloco atual, como demonstrado pelas linhas vermelhas da Figura 9 (a). Caso a predição fosse realizada desta forma, o bloco predito seria

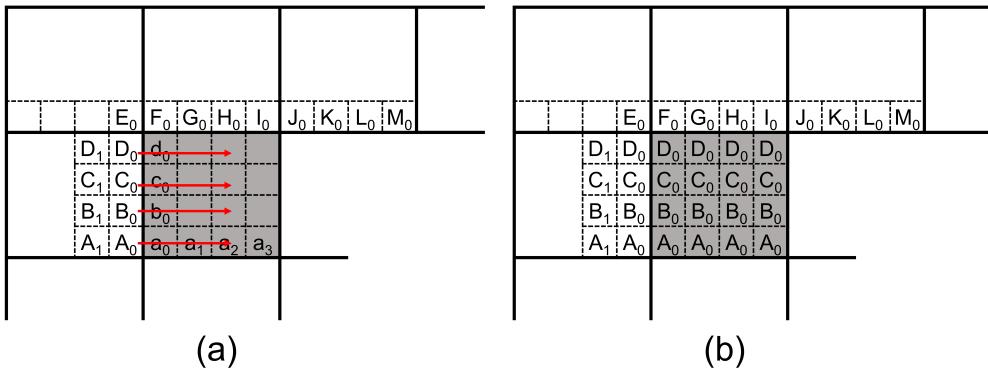


Figura 9 – Exemplo de funcionamento da predição intraquadro. (a) bloco antes da predição, (b) bloco após a predição

representado como demonstrado na Figura 9 (b). Também é possível realizar a predição intraquadro utilizando-se técnicas mais elaboradas, como considerar todas as amostras do bloco vizinho (desde  $A_0, B_0, C_0, D_0$  até  $A_3, B_3, C_3, D_3$ ) para realizar uma extração de maior ordem. De modo geral, o número de amostras consideradas e o cálculo da predição é dependente do padrão de codificação sendo empregado.

Os padrões de codificação de vídeo modernos empregam modos de predição direcionais com múltiplos ângulos de orientação. Caso um modo de predição direcional vertical fosse usado, o bloco atual seria predito com base nas amostras de  $F_i$  a  $I_i$ , e caso um modo diagonal fosse empregado, a predição iria considerar uma combinação ponderada de diversas amostras de acordo com o ângulo de inclinação. Os modos de predição não-direcionais não serão abordados nesta seção visto que estes são dependentes do padrão de codificação sendo empregado. Ao invés disso, os modos de predição não-direcionais do padrão HEVC serão discutidos na Seção 3.3.

Já para explorar a redundância temporal (entre quadros vizinhos) os padrões de codificação de vídeo utilizam a predição interquadros, que busca representar o bloco a ser predito com base em blocos já preditos em outros quadros. Através da predição interquadros é possível obter uma predição compensada por movimento (*motion-compensated prediction* - MCP) para cada bloco do quadro atual (CONCEIÇÃO, 2016).

A MCP se baseia no modelo de movimento translacional (SZE; BUDAGAVI; SULLIVAN, 2014), isto é, considera que os objetos da cena não sofrem rotação ou movimentos de aproximação/distanciamento da câmera. Para realizar a predição de um determinado bloco, a MCP consiste em comparar o bloco a ser predito com quadros já codificados à procura de uma região que seja tão próxima do bloco a ser predito quanto possível. Esta técnica de procurar o bloco a ser predito dentro de quadros já codificados é chamada de algoritmo de casamento de blocos (*block matching algorithm* - MBA). Existem três parâmetros principais que regem o funcionamento de

qualquer MBA: o tamanho da área de busca, o número de quadros de referência e o critério de similaridade.

Cada quadro cujos blocos são comparados ao bloco a ser predito durante o MBA é chamado de quadro de referência. Os quadros de referência podem estar tanto temporalmente à frente quanto atrás do quadro do bloco a ser predito. Além disso, é possível que o bloco a ser predito seja representado através de uma combinação linear de blocos de múltiplos quadros (SZE; BUDAGAVI; SULLIVAN, 2014).

A área de busca corresponde à região dentro do quadro de referência que será avaliada à procura da melhor correspondência com o bloco a ser predito. O tamanho da área de busca é variável, podendo até mesmo assumir o tamanho total do quadro. Visto que a taxa de amostragem de quadros de vídeos digitais é elevada (a partir de 24 quadros por segundo) e os quadros de referência costumam estar temporalmente próximos ao quadro a ser predito, é natural que exista pouca variação entre o quadro a ser predito e o de referência. Sendo assim, costuma-se avaliar apenas uma porção do quadro de referência (a área de busca) para não desperdiçar recursos computacionais durante a codificação.

Já o critério de similaridade indica qual métrica é utilizada para comparar o bloco a ser predito e o bloco candidato dentro da área de busca. Existem múltiplos critérios de similaridade disponíveis, alguns mais elaborados do que outros, mas os mais comumente utilizados são a Soma das Diferenças Absolutas (*Sum of Absolute Differences* - SAD) e o Erro Médio Quadrático (*Mean Squared Error* - MSE).

Embora o MBA possa estar restrito a uma área de busca dentro dos quadros de referência, esta busca não se limita à posições inteiras. Visando representar o movimento de forma mais precisa e suave, a predição interquadros permite que o bloco a ser predito seja comparado com posições fracionárias dentro do quadro de referência através da estimativa de movimento fracionária (*fractional motion estimation* - FME) (SULLIVAN et al., 2012). Para isso, é empregado um filtro de interpolação sobre o quadro de referência para derivar as amostras intermediárias, permitindo que o MV resultante possua precisão fracionária.

Uma vez que tenha sido encontrada a melhor correspondência para o bloco a ser predito dentro dos quadros de referência, é gerado um vetor de movimento (*motion vector* - MV) que caracteriza o movimento vertical e horizontal do bloco de referência em relação ao bloco atual (CONCEIÇÃO, 2016).

Um diagrama representando a predição interquadros é apresentado na Figura 10. Nesse caso, o bloco a ser predito é representado pelo bloco hachurado claro no quadro atual (o mais à direita na Figura 10 (a)). Esse bloco será procurado dentro da área de busca dos quadros de referência, delimitada pelas linhas tracejadas dentro destes. Uma vez avaliada a área de busca dentro dos quadros de referência, o BMA determina qual bloco já codificado possui a melhor correspondência com o bloco a ser

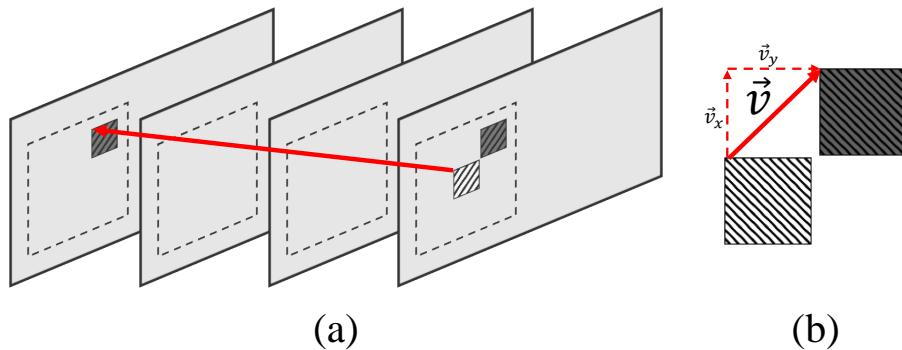


Figura 10 – Exemplo de funcionamento da predição interquadros

predito, e define um MV  $\vec{v}$ , caracterizado pela linha vermelha na Figura 10 (a). O MV  $\vec{v}$  possui duas componentes,  $v_x$  e  $v_y$ , que representam o deslocamento horizontal e vertical, respectivamente, do bloco a ser predito em relação ao bloco de referência. A decomposição de  $\vec{v}$  em  $v_x$  e  $v_y$  é apresentada na Figura 10 (b).

Após realizar a predição do bloco atual o bloco resultante é uma aproximação do bloco original. Esta aproximação pode ser mais ou menos precisa de acordo com a predição e referências utilizadas, entretanto, raramente o bloco original e predito possuem uma correspondência exata.

Sendo assim, após a predição os blocos preditos são somados a um bloco residual para que possam corresponder ao bloco original. O bloco residual é calculado de forma direta ao subtrair-se o bloco predito do bloco original, e representa o que deve ser adicionado ao bloco predito para que este corresponda ao bloco original. Sendo assim, tomando como exemplo a predição interquadros apresentada na Figura 10, o bloco original pode ser caracterizado pelo bloco co-localizado no quadro de referência, deslocado pelo MV  $\vec{v}$  e somado ao bloco residual.

### 3.2 Tratamento de Resíduo

Embora a predição intraquadro e interquadros torne possível representar um bloco com base em outros blocos e o erro de predição, uma observação atenta mostra que esse processo gera mais dados do que existiam originalmente: o erro de predição possui as mesmas dimensões que o bloco original, e além dele é necessário salvar o modo de predição que foi utilizado. Embora esse processo de fato aumente a quantidade de dados, ele prepara os dados para que sejam comprimidos de forma mais eficiente como será demonstrado nos próximos parágrafos.

Como a predição – tanto intraquadro como interquadros – dos padrões de codificação atuais é bastante eficiente, o bloco original e predito costumam ser muito semelhantes, fazendo com que o erro de predição seja composto principalmente por zeros e valores de pequena magnitude. Esse erro de predição é então transformado

para ir do domínio do espaço para o domínio da frequência.

Como o bloco residual é composto principalmente por valores zero e de baixa magnitude, a variação espacial entre os valores é bastante suave. Dado este comportamento, o bloco transformado tende a apresentar coeficientes de baixa frequência com alta magnitude e coeficientes de alta frequência com magnitude pequena ou até mesmo zero.

Como o sistema visual humano é pouco sensível às altas frequências, é possível representar os coeficientes de alta frequência de forma aproximada, ou até mesmo eliminá-los, sem causar uma degradação visível na imagem. A quantização é a etapa responsável por fazer esta aproximação: para um dado conjunto de dados representando a magnitude dos coeficientes, a quantização é responsável por mapear este conjunto de dados para um conjunto de dados menor, de forma que quando esta operação é desfeita uma parcela dos dados é perdida. A maneira mais simples de representar a quantização é através da divisão inteira dos coeficientes da transformada por um valor de quantização, de forma que a parte decimal da divisão é perdida. Como o conjunto de dados na saída da quantização é menor do que na entrada desta, a quantização aumenta a redundância na representação do vídeo, tornando este mais propício a ser comprimido.

Por fim, uma vez que os dados estão num estado ainda mais redundante, a codificação entrópica é utilizada. A codificação entrópica cria uma nova representação dos dados em que os símbolos que ocorrem com maior frequência, como zeros e valores de pequena magnitude, são representados com menos bits, enquanto que símbolos que ocorrem com menor frequência, como valores de alta magnitude, são representados com mais bits. Como após o processo de transformada e quantização a maior parte dos coeficientes são nulos e de pequena magnitude, a codificação entrópica consegue representar a maior parte do vídeo com poucos bits, atingindo assim um elevado grau de compressão. Para realizar a codificação entrópica, é possível utilizar técnicas baseadas em codificação aritmética (WITTEN; NEAL; CLEARY, 1987) e Huffman (HUFFMAN, 1952).

### 3.3 O Padrão HEVC

O padrão HEVC é um dos mais recentes padrões de codificação de vídeo e sucessor do padrão H.264 *Advanced Video Coding* (AVC) (JVT, 2003) que dominou o mercado por vários anos. O padrão HEVC herdou e melhorou uma série de ferramentas dos padrões anteriores, além de incorporar novas ferramentas. Esse conjunto de ferramentas fez com que o padrão HEVC apresente uma taxa de compressão em média 40% maior do que o padrão H.264 AVC para uma mesma qualidade visual (OHM et al., 2012).

O padrão HEVC apresenta uma abordagem de codificação híbrida, como a maioria dos padrões de codificação de vídeo atuais. Essa abordagem híbrida consiste em realizar a predição dos quadros através de uma predição intraquadro (espacial) ou interquadros (temporal), realizar uma transformada bidimensional (2D) sobre os blocos preditos e aplicar uma codificação de entropia sobre os coeficientes quantizados.

Entretanto, até que uma determinada região do vídeo seja codificada de forma definitiva é necessário que inúmeras opções de codificação sejam avaliadas para determinar qual delas apresenta a maior eficiência de codificação e isto é realizado através de um processo chamado *Rate-Distortion Optimization* (RDO). O RDO é um método desenvolvido para otimizar uma função que considera a taxa de bits (Rate) e a qualidade visual do vídeo (Distortion), ou seja, é um método que determina com exatidão e alto custo computacional qual método de codificação resulta no melhor custo-benefício nos aspectos taxa de bits e qualidade visual (SULLIVAN; WIEGAND, 1998). Sendo assim, o RDO consiste em avaliar vários métodos de codificação para uma mesma região do quadro, isto é, diferentes tamanhos de bloco, vetores de movimento, modos de predição intraquadro, tamanhos de transformada, entre outros, e determinar qual destas combinações resulta na melhor relação entre a qualidade (distorção) e a taxa de *bits* (taxa) necessária para representar tal região. Sendo assim, primeiro avalia-se a eficiência de codificar esta região com todos os modos de predição e tamanhos de transformada possível, depois avalia-se a eficiência de dividir esta região em regiões menores e codificar cada uma delas com todos os métodos de codificação disponíveis, em seguida cada região é subdividida novamente e o processo é repetido até que o menor tamanho de bloco possível seja atingido.

O RDO é realizado para cada método de codificação de acordo com a Equação 1, onde  $D$  representa a distorção causada por um determinado método de codificação,  $R$  representa a taxa necessária para representar tal método de codificação,  $J$  representa o custo RD deste método de codificação e pode ser considerado como a aptidão deste método de codificação para a região em questão, e  $\lambda$  representa um multiplicador lagrangiano que prioriza uma variável em detrimento da outra (SULLIVAN; WIEGAND, 1998).

$$J = D + \lambda \times R \quad (1)$$

No padrão HEVC, a variável  $\lambda$  é derivada de um parâmetro de codificação chamado *Parâmetro de Quantização* (*Quantization Parameter* - QP), que é utilizado para controlar se o codificador deve produzir um vídeo com grande qualidade (baixa distorção) ou baixa taxa de *bits*. Sendo assim, QPs maiores priorizam a taxa de compressão em detrimento da qualidade do vídeo, enquanto que QPs menores dão menos prioridade à taxa de compressão e levam a melhores qualidades visuais.

O RDO apresenta uma grande complexidade dentro do codificador, visto que ele precisa realizar todos os passos da codificação para determinar os valores de  $D$  e  $R$ . Para um determinado modo de predição e tamanho de bloco, o codificador realiza a predição das amostras, subtrai do bloco original, em seguida este bloco de resíduo deve ser transformado e quantizado, para depois disso, ser reconstruído e calcular a distorção  $D$ . Já para o cálculo de  $R$ , é necessário realizar a codificação entrópica após a quantização para obter a taxa de compressão atingida.

As sub-seções a seguir explicam de forma mais detalhada como as diferentes etapas de particionamento e predição do padrão HEVC são implementadas.

### 3.3.1 Configurações de Codificação e Estruturas de Particionamento

O padrão HEVC é composto por três configurações de codificação, cada uma delas desenvolvida para atingir um conjunto de requisitos. Estas configurações são *All Intra*, *Low Delay* e *Random Access*. Quando a codificação é realizada com a configuração *All Intra*, todos os quadros do vídeo são codificados utilizando exclusivamente a predição intraquadro, ou seja, não existem dependências entre os quadros codificados. Como a configuração *All Intra* utiliza apenas a predição intraquadro, esta apresenta uma baixa complexidade e baixa taxa de compressão.

Quando a configuração *Low Delay* é utilizada, o primeiro quadro do vídeo é codificado com a predição intraquadro, enquanto que os quadros seguintes são codificados com a predição interquadros, fazendo com que um quadro ao meio do vídeo possua dependências em relação a todos os quadros anteriores. A configuração *Low Delay* codifica os quadros na mesma ordem em que estes vão ser exibidos ao usuário e apresenta uma boa taxa de compressão, além disso ela costuma ser usada em aplicações que demandem tempo real – daí o nome *Low Delay*.

Já a configuração *Random Access* é semelhante à configuração *Low Delay*. Quando a configuração *Random Access* é utilizada, o codificador emprega alguns ponto de ancoragem durante a codificação através de *intraperiods*. Um *intraperiod* determina quantos quadros serão codificados com a predição interquadros até que um quadro seja codificado exclusivamente com a predição intraquadro. Essa técnica faz com que um quadro ao meio do vídeo possua dependências com apenas alguns poucos quadros, aumentando a robustez de transmissões de vídeo. Além disso, a configuração *Random Access* não codifica os quadros na mesma ordem em que estes são apresentados ao usuário. Ao invés disso, dentro de um *intraperiod* os quadros são codificados numa ordem de vai-e-vem, o que melhora o resultado da predição em relação à oclusões e descobrimentos, melhorando a eficiência de codificação.

A principal inovação do padrão HEVC em relação aos padrões anteriores se dá na sua estrutura de particionamento altamente flexível. Cada quadro do vídeo é dividido em blocos básicos denominados *Coding Tree Units* (CTUs), que são compostos por

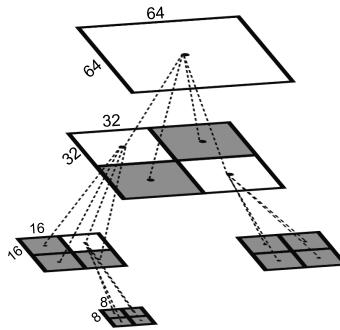


Figura 11 – Subdivisão de uma CTU em várias CUs. Adaptado de (CORREA et al., 2015)

um bloco de luminância e um bloco para cada crominância. Além disso, o padrão HEVC prevê que as CTUs possam ter dimensões desde  $16 \times 16$  até  $64 \times 64$  (SULLIVAN et al., 2012).

Cada uma destas CTUs é a raiz de uma árvore quadrática (*quadtree* - QT). Essa árvore pode se subdividir recursivamente até blocos de tamanho  $8 \times 8$ , denominados *Coding Units* (CUs), de forma que as CUs podem ter dimensões desde  $8 \times 8$  até  $64 \times 64$  – caso em que uma CTU não é subdividida e dá origem a apenas uma CU (SULLIVAN et al., 2012). Um exemplo de uma CTU sendo sub-dividida em CUs é apresentado na Figura 11, onde os blocos acinzentados representam as folhas da árvore (CUs que não se dividem novamente) e os blocos brancos representam os nós internos da árvore (CUs que foram subdivididas).

Cada CU é dividida novamente, dando origem a *Prediction Units* (PUs). O tipo de predição (intraquadro ou interquadros) a ser utilizado é definido a nível de CU, de modo que todas as PUs pertencentes a uma CU devem passar pelo mesmo tipo de predição (SULLIVAN et al., 2012). Quando uma CU é codificada com a predição intraquadro, cada CU dá origem a uma PU com as mesmas dimensões. A única exceção ocorre em CUs de dimensão  $8 \times 8$ , que podem dar origem a quatro PUs de dimensões  $4 \times 4$  (SULLIVAN et al., 2012).

Quando uma CU é codificada com a predição interquadros, esta pode se subdividir de várias formas diferentes. Para explicar estas divisões, supõe-se que uma CU é composta por  $M \times M$  amostras e a Figura 12 será usada como exemplo.

Dessa forma, uma CU pode dar origem a apenas uma PU com dimensões  $M \times M$ , ou dar origem a duas PUs em múltiplos particionamentos distintos. Uma das possibilidades é dividir a CU em duas PUs retangulares com mesmas dimensões, dando origem a duas PUs  $M \times M/2$  ou  $M/2 \times M$ .

A outra possibilidade é dividir a CU em duas PUs retangulares assimétricas, denominadas *asymmetric motion partitioning* (AMP). Existem quatro particionamentos

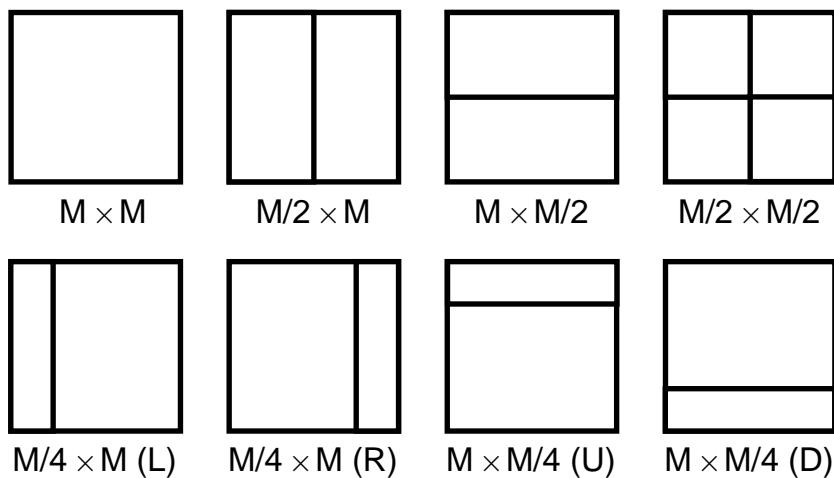


Figura 12 – Modos de particionamento de CUs em PUs. Adaptado de (SULLIVAN et al., 2012)

AMP disponíveis:  $M/4 \times M$ , que pode ter a PU menor na esquerda ( $M/4 \times M$  L) ou na direita ( $M/4 \times M$  R), e  $M \times M/4$ , que pode ter a PU menor na parte superior ( $M \times M/4$  U) ou inferior( $M \times M/4$  D) (SULLIVAN et al., 2012). Vale ressaltar que as PUs AMP são permitidas apenas para CUs  $16 \times 16$  ou maiores.

O modo  $M/2 \times M/2$  apresentado na Figura 12 caracteriza as CUs  $8 \times 8$  que foram divididas em PUs  $4 \times 4$  durante a predição intraquadro.

Cada CU também dá origem a um conjunto de *Transform Units* (TUs). Essa divisão se dá através de uma árvore denominada *Residual Quadtree* (RQT), sendo que as TUs podem ter dimensões desde  $32 \times 32$  até  $4 \times 4$  amostras (SZE; BUDAGAVI; SULLIVAN, 2014). As TUs são utilizadas durante a codificação residual, isto é, o erro de predição é dividido entre as TUs e transformado. Diferente dos padrões anteriores, o padrão HEVC permite que as TUs compreendam mais de uma PU para maximizar os benefícios da divisão em RQT (SULLIVAN et al., 2012).

### 3.3.2 Predição Intraquadro

O padrão HEVC possui 35 modos de predição intraquadro, sendo que 33 destes são modos direcionais, enquanto que dois são modos não-direcionais. Os dois modos não-direcionais são denominados Planar e DC (SULLIVAN et al., 2012).

Os modos angulares são representados pelos números desde 2 até 34, enquanto que os modos Planar e DC são representados pelos números 0 e 1, respectivamente, como demonstrado na Figura 13. Os modos direcionais foram projetados de forma que cubram de forma mais densa os ângulos próximos de  $0^\circ$  e  $90^\circ$ , e cubram de forma mais esparsa os ângulos diagonais. Esta distribuição busca explorar a tendência dos objetos na cena em apresentar formas verticais e horizontais, e o ângulo de predição exato de cada modo está disponível em (JCT-VC, 2013).

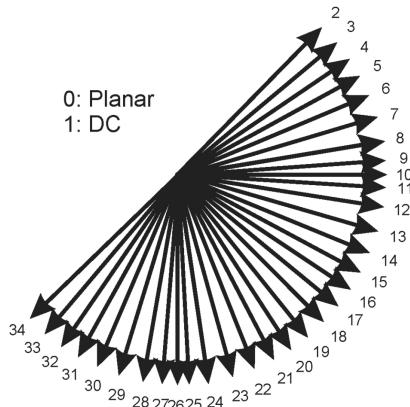


Figura 13 – Modos de predição intraquadro (SULLIVAN et al., 2012)

Uma codificação intraquadro utilizando qualquer um dos modos direcionais funciona de acordo com o esquema apresentado na Figura 9, em que as amostras da PU atual são representadas através de extração das amostras vizinhas. Já os modos Planar e DC possuem um funcionamento distinto, próprio do padrão HEVC.

O modo de predição DC normalmente é empregado em PUs que apresentem uma textura muito homogênea. Para isso, o modo DC representa todas as amostras dentro de uma PU utilizando a média entre as amostras da PU. Como o modo DC representa uma região quadrada da imagem com valores exatamente iguais, isso pode criar descontinuidade entre a PU atual e as PUs vizinhas. O modo de predição Planar é semelhante ao modo DC, mas atenua o problema da continuidade entre blocos vizinhos. Ao invés de usar a média, o modo Planar avalia os quatro cantos da PU e realiza duas predições lineares, e as amostras dentro da PU são representadas pela média entre estas duas predições amostra-a-amostra (SULLIVAN et al., 2012). Com esse comportamento, o modo Planar é capaz de representar regiões homogêneas com variações suave de textura/iluminação.

Os padrões de codificação de vídeo definem o que pode ser feito durante a codificação, como por exemplo, quais particionamentos são permitidos, quais quadros de referência podem ser usados, quais modos de predição podem ser utilizados em quais situações, entre outros, e como as decisões da codificação devem ser sinalizadas no *bitstream*. Entretanto, o padrão não define como as decisões de codificação devem ser tomadas: é possível fazer avaliações exaustivas ou empregar heurísticas.

Ainda assim, o padrão HEVC possui um *software* de referência que implementa o codificador HEVC, denominado *HEVC Test Model* (HM) (JCT-VC, 2017). Esse *software* implementa todas as ferramentas do padrão HEVC, e possui uma abordagem mista entre testes exaustivos e heurísticas. Além disso, o HM costuma ser utilizado na literatura para comparar o desempenho entre diversas técnicas propostas.

Como avaliar os 35 modos de predição intraquadro para todos os tamanhos de PU possíveis é uma tarefa computacionalmente muito custosa, o HM emprega uma

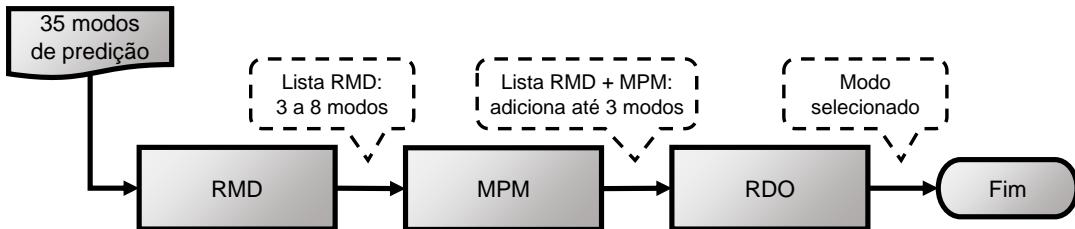


Figura 14 – Fluxo de Processamento da Predição Intraquadro

abordagem em três etapas para avaliar os modos de predição intraquadro. Essa abordagem consiste em realizar *Rough Mode Decision* (RMD), seguido da derivação dos *Most Probable Modes* (MPMs), e por fim *Rate-Distortion Optimization* (RDO) (JCT-VC, 2017). Um esquema da predição intraquadro é apresentado na Figura 14.

Primeiro é realizado o RMD, que consiste num algoritmo de baixo custo (aproximação do RDO) para classificar os modos de predição que provavelmente deem os melhores resultados. Para obter essa estimativa, a PU atual é codificada com os 35 modos de predição intraquadro, e o *Sum of Absolute Transformed Differences* (SATD) é aplicado a cada uma destas 35 predições. O SATD consiste em aplicar uma Transformada Hadamard sobre os resíduos de predição e em seguida somar estes resíduos, de forma que o RMD age como uma aproximação do custo RDO. Após calcular o SATD para os 35 modos, os **N** modos com o menor SATD são colocados numa lista de melhores modos candidatos. Para PUs  $64 \times 64$ ,  $32 \times 32$  e  $16 \times 16$  **N** assume o valor 3, enquanto que para PUs  $8 \times 8$  e  $4 \times 4$  **N** assume o valor 8 (JCT-VC, 2017). Essa distinção se deve ao fato de que PUs maiores tendem a apresentar texturas mais homogêneas, logo, não é necessário avaliar tantos modos para obter bons resultados (KHAN; SHAFIQUE; HENKEL, 2013).

Após o RMD, a derivação dos MPMs busca explorar a correlação entre PUs espacialmente vizinhas. Para isso, duas PUs vizinhas já codificadas são avaliadas, e com base nos modos de predição intraquadro destas PUs, mais três modos são adicionados à lista de melhores modos candidatos. É importante ressaltar que não são colocados modos repetidos na lista de melhores modos candidatos: caso algum modo derivado como MPM já tenha sido adicionado na lista durante o RMD, ele é mantido na lista e sua réplica é descartada (JCT-VC, 2017). A Figura 15 exemplifica a PU a ser predita  $P$ , e as duas PUs vizinhas,  $P_A$  e  $P_B$ , avaliadas para derivar os MPMs.

Após o RMD e a derivação dos MPMs, a lista de melhores modos candidatos pode ter de 3 a 11 modos. A lista possui 3 modos quando a PU sendo predita tem dimensões  $64 \times 64$ ,  $32 \times 32$  ou  $16 \times 16$ , e os 3 modos selecionados pelo RMD são os mesmos do MPM. A lista possui 11 modos quando a PU sendo predita tem dimensões  $8 \times 8$  ou  $4 \times 4$ , e os 3 modos derivados como MPMs são diferentes dos modos já selecionados pelo RMD. Listas com quantidades de modos intermediárias ocorrem de acordo

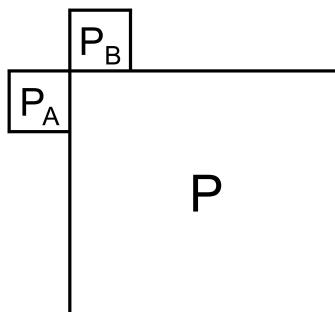


Figura 15 – PU atual e vizinhas na derivação dos MPMs

com o tamanho da PU sendo predita e a redundância entre as decisões do RMD e MPM. Vale ressaltar que esta lista nunca terá 7 modos, visto que o maior número de modos avaliados para as PUs maiores é 6, enquanto que o menor número de modos avaliados para PUs menores é 8.

Depois disso, a lista de melhores modos candidatos é encaminhada ao RDO para avaliar qual destes modos apresenta o melhor custo-benefício. Após a etapa de RDO, a predição intraquadro sabe com certeza qual dos modos pertencentes à lista de modos candidatos apresenta a melhor eficiência de codificação. É importante lembrar que neste caso o RDO é realizado de maneira local, para decidir entre os modos de predição intraquadro. Em um panorama geral, o RDO também é utilizado para escolher entre as predições e tamanhos de bloco.

### 3.3.3 Predição Interquadros

Caso o codificador empregue a predição interquadros, o conjunto de quadros de referência está relacionado à configuração de codificação sendo utilizada. Caso a configuração *Low Delay* esteja sendo utilizada, o conjunto de quadros de referência será composto exclusivamente por quadros temporalmente anteriores ao quadro sendo codificado. Já se a configuração *Random Access* estiver sendo utilizada, o conjunto de quadros de referência pode ser composto por quadros que estão temporalmente tanto à frente quanto atrás do quadro sendo codificado.

O padrão HEVC suporta dois tipos de predição interquadros: a uni-predição e a bi-predição. A uni-predição ocorre quando apenas um conjunto de dados de movimento é utilizado na geração da MCP, ou seja, apenas um MV é gerado para o bloco a ser predito. Já a bi-predição ocorre quando dois conjuntos de dados de movimento são utilizados na geração da MCP, ou seja, são gerados dois MVs. Esses dados de movimento podem ser provenientes de dois quadros distintos, e a representação final do bloco predito é baseado nestes dois MVs (SZE; BUDAGAVI; SULLIVAN, 2014).

No software de referência do padrão HEVC, a predição interquadros é composta por três módulos principais: um algoritmo de casamento de blocos (MBA) denominado

*Test Zone Search* (TZS), uma estimativa de movimento fracionária para refinar os vetores provenientes do TZS, e dois modos de predição especiais que reutilizam a informação de movimento de blocos vizinhos. Cada um destes módulos será discutido nos próximos parágrafos.

- *Test Zone Search*

No software de referência do padrão HEVC, o algoritmo de casamento de blocos (MBA) *Test Zone Search* (TZS) é utilizado, visto que este é um algoritmo que apresenta excelentes resultados em termos de taxa de compressão e qualidade visual enquanto apresenta uma complexidade computacional significativamente menor do que realizar uma busca exaustiva dentro da área de busca. O TZS é composto por quatro etapas, como demonstrado na Figura 16.

Num primeiro momento, a etapa de **Predição de Vetores de Movimento** é responsável por avaliar um conjunto de informações de movimento derivadas a partir de regras pré-definidas. Nesta etapa, os vetores de movimento gerados são avaliados para determinar qual deles apresenta a melhor eficiência de codificação, para nas próximas etapas, realizar mais buscas em torno deste vetor. Dessa forma, a **Predição de Vetores de Movimento** é capaz de acelerar a busca realizada nas próximas etapas visto que direciona as buscas para um ponto onde existe grande probabilidade de encontrar-se o melhor bloco.

Após a **Predição de Vetores de Movimento** é realizada a **Busca Inicial**, que consistem em realizar uma busca em expansão ao redor do ponto apontado pelo vetor de movimento predito anteriormente. Esta busca em expansão pode ser realizada de duas formas distintas no HM, a busca em diamante e a busca quadrada, que são apresentadas na Figura 17 (a) e (b), respectivamente. Na Figura 17 a busca é realizada dentro de um área de busca de  $16 \times 16$  pixels. Tanto na busca em diamante quanto quadrada, os círculos indicam a posição da aresta do bloco que está sendo avaliado, enquanto que o número indica o nível da busca em relação ao ponto inicial. Nos níveis 1, 2, 4 e 8 são avaliados oito blocos que estão às distâncias de 1, 2, 4 e 8 pixels do ponto central, respectivamente. Para os pontos a partir do nível dezesseis, são

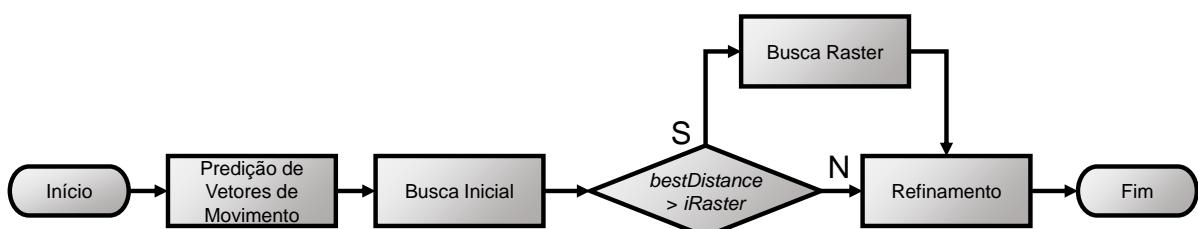


Figura 16 – Fluxograma do algoritmo *Test Zone Search*

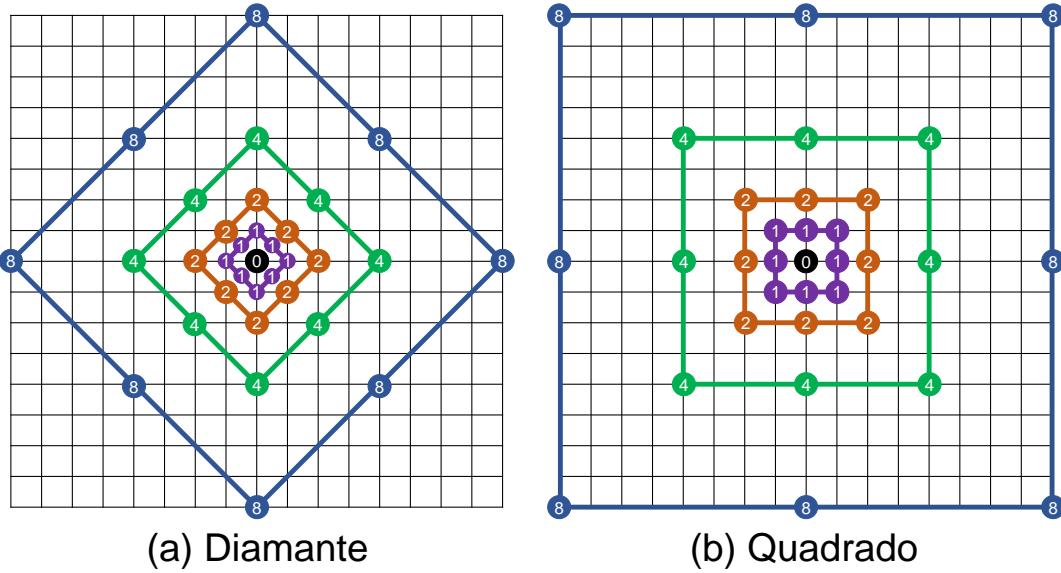


Figura 17 – Padrão de busca da Busca Inicial no TZS. (a) Diamante e (b) quadrada. Adaptado de (GONÇALVES, 2018)

avaliados dezesseis blocos por nível. Além disso, como demonstrado pela Figura 17, a distância dos blocos avaliados cresce de forma exponencial em relação ao ponto central da busca: primeiro são avaliados bloco à distância de 1 pixel ( $2^0$ ), depois à distância de 2 pixels ( $2^1$ ), depois 4 ( $2^2$ ) e 8 ( $2^3$ ), respectivamente. Esta busca é realizada até que o limite da área de busca seja alcançada, ou após avaliar três níveis consecutivos sem encontrar um bloco melhor do que o bloco atual.

Caso o melhor bloco proveniente da **Busca Inicial** esteja a uma distância maior do que uma constante  $iRaster$  – parâmetro do algoritmo TZS – a etapa de **Busca Raster** é realizada. Já se a distância for menor do que a constante  $iRaster$ , então a **Busca Raster** não é realizada e o Refinamento é executado diretamente.

A etapa de **Busca Raster** do TZS é muito semelhante a uma análise exaustiva, onde os blocos dentro da área de busca são avaliados um a um. A principal diferença é que na **Busca Raster** esta avaliação é feita de forma subamostrada de acordo com a constante  $iRaster$ , de forma que ao invés de avaliar todos os blocos no sentido horizontal e vertical, apenas um em cada  $iRaster$  blocos é avaliado tanto horizontalmente quanto verticalmente. A única condição de parada da Busca Raster é o tamanho da área de busca, de forma que mesmo que um bloco melhor do que o atual seja encontrado nas primeiras avaliações, toda a área de busca é avaliada. Caso o vetor de movimento da **Busca Inicial** seja muito grande (maior do que  $iRaster$ ) existe uma grande possibilidade de que os blocos avaliados até então não foram bons candidatos, de forma que o objetivo da **Busca Raster** é avaliar um grande conjunto de blocos para conduzir as buscas para uma região mais adequada novamente.

Por fim, a etapa de **Refinamento** tem o objetivo de refinar a informação de movimento encontrada nas etapas anteriores. Esta etapa funciona de forma semelhante à

**Busca Inicial**, com algumas adaptações. Dado um vetor de movimento proveniente da **Busca Inicial** ou da **Busca Raster**, a etapa de **Refinamento** faz uma nova busca em expansão ao redor deste vetor de movimento, com um passo de crescimento de  $2^n$ . Entretanto, caso um bloco melhor seja encontrado a etapa de **Refinamento** é iniciada com passo um novamente, onde o ponto central é o melhor bloco da última iteração. Esse processo de avaliar, expandir e reiniciar a busca é realizada até que se atinja o limite da área de busca ou sejam feitas duas expansões sem que um melhor bloco seja encontrado. Isto é, quando a busca é reiniciada em torno de um bloco, se nenhum novo melhor bloco for encontrado até o final da segunda expansão, a busca é encerrada.

De acordo com a descrição do funcionamento do TZS e em diversas análises presentes na literatura (GONÇALVES et al., 2017), fica claro que a etapa de **Busca Raster** é responsável pela maior parte da complexidade computacional presente no algoritmo de TZS.

- Estimação de Movimento Fracionária

Após definir o melhor vetor de movimento inteiro para o bloco atual, a estimação de movimento fracionária (*fractional motion estimation* - FME) é realizada para representar movimentos mais suaves. O padrão HEVC suporta precisões de meio pixel e de um quarto de pixel (SULLIVAN et al., 2012), e o funcionamento da FME será apresentado com o auxílio da Figura 18.

Na Figura 18, a seta  $MV_0i$  representa o vetor de movimento antes de realizar a FME, a seta  $MV_1$  representa o vetor de movimento após realizar a FME, os círculos vermelhos representam as posições de pixels inteiras, os quadrados azuis representam as posições de pixels fracionários com precisão de  $1/2$  pixel e os triângulos verdes representam as posições de pixels fracionários com precisão de  $1/4$  pixel.

Antes da FME a estimação de movimento gera um MV que aponta para uma posição inteira, apresentada no *Passo 0* da Figura 18. Dessa forma, o *Passo 1* da FME é gerar as oito amostras fracionárias com precisão de  $1/2$  pixel ao redor do MV original, e avaliar qual destes MVs com precisão de  $1/2$  pixel obtém a melhor eficiência de codificação. Supondo que a amostra que leve aos melhores resultados seja a no canto inferior direito, então é gerado um vetor de movimento fracionário  $MV_0h$  no *Passo 2* que representa o movimento da posição inteira para a posição fracionária com precisão de  $1/2$  pixel. Feito isso, o *Passo 3* da FME é gerar as oito amostras fracionárias com precisão de  $1/4$  pixel ao redor do melhor MV atualizado ( $MV_0i+MV_0h$ ) e avaliá-las para determinar qual propicia a melhor eficiência de codificação. Caso a amostra com precisão de  $1/4$  pixel que retorne a melhor eficiência de codificação seja a esquerda-central, então é gerado um novo vetor de movimento fracionário  $MV_0q$  no *Passo 4*. Por fim, no *Passo 5* o vetor de movimento inteiro  $MV_0i$  e os dois vetores de movimento

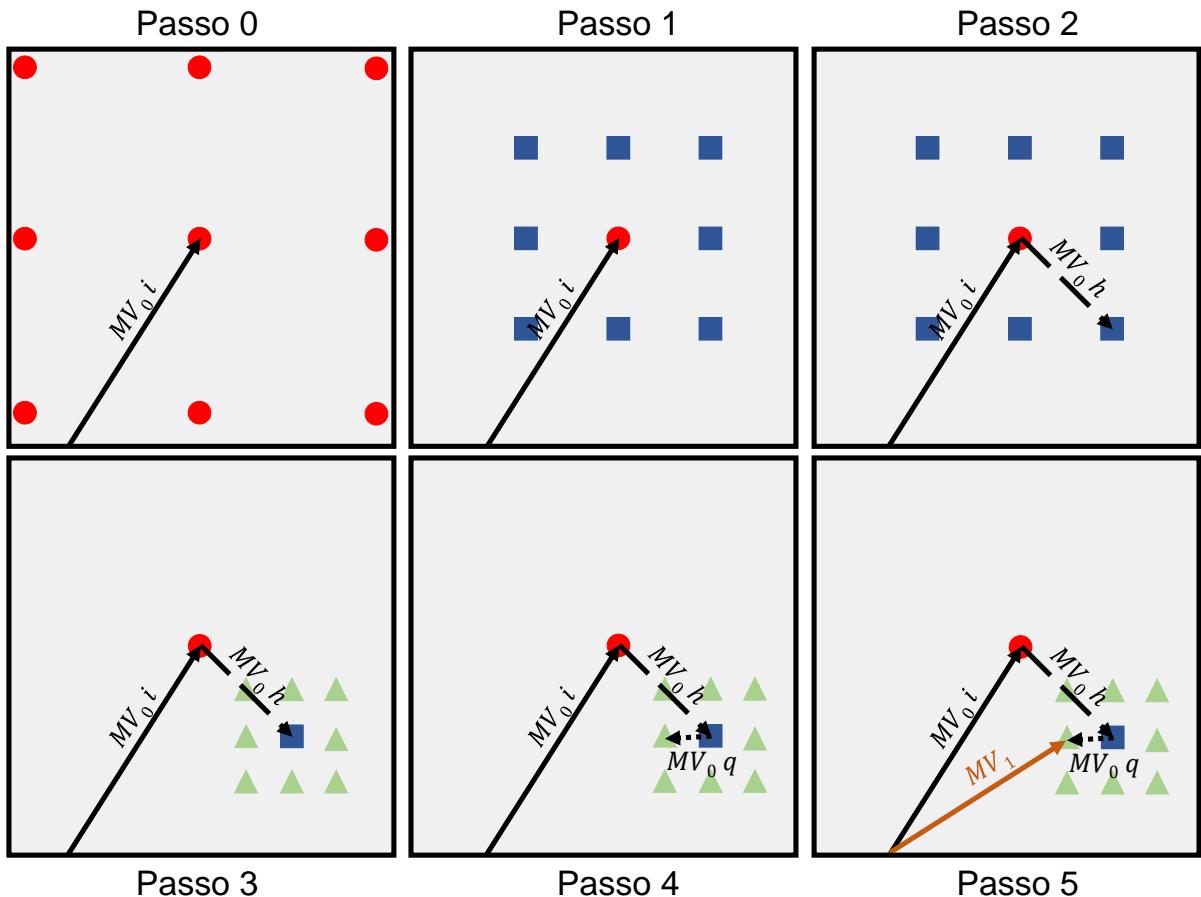


Figura 18 – Funcionamento da estimação de movimento fracionária

fracionários  $MV_0h$  e  $MV_0q$  são somados, dando origem ao vetor de movimento final  $MV_1$

- Modos especiais *merge* e *skip*

Além da representação da estimação de movimento convencional, onde um bloco é predito através de um MV – tanto inteiro quanto fracionário – que indica o deslocamento deste bloco em relação a uma referência, o padrão HEVC prevê dois modos especiais de predição interquadros: os modos *merge* e *skip*.

A codificação de vídeos utilizando particionamento em blocos é muito utilizada pois possui muitos benefícios, como utilizar diferentes modos de predição para regiões distintas do vídeo, além de poder investir mais recursos computacionais em regiões de maior interesse. Entretanto, essa divisão em blocos pode separar regiões correlacionadas do vídeo, que poderiam compartilhar do mesmo modo de predição caso estivessem encapsuladas no mesmo bloco (SZE; BUDAGAVI; SULLIVAN, 2014). Os modos *merge* e *skip* foram desenvolvidos para contornar este problema da codificação de vídeo baseada em blocos.

Para avaliar o modo *merge*, primeiro é gerada uma lista de candidatos *merge* que corresponde a todos os blocos dos quais os dados de movimento podem ser herdados. Os candidatos *merge* são derivados a partir de blocos que já foram codificados, e estes blocos podem ser vizinhos tanto espaciais quanto temporais do bloco atual. A partir destes candidatos, novos candidatos *merge* são gerados a partir de combinações entre os candidatos já presentes na lista e seguindo um conjunto de regras pré-determinado.

Como a lista de candidatos é gerada a partir de blocos já codificados, ela pode ser gerada de forma idêntica tanto no codificador quanto no decodificador. Isso permite que, ao invés de sinalizar um vetor de movimento de forma explícita para o bloco atual, apenas um índice correspondente ao candidato do qual o movimento foi herdado seja sinalizado. Desta forma, o modo *merge* possibilita que um conjunto de blocos que compartilham as mesmas informações de movimento sejam sinalizados através de um vetor de movimento explícito e uma série de referências para este vetor, atingindo uma elevada eficiência de codificação. Embora as informações de movimento sejam sinalizadas de forma compacta, o resíduo de predição ainda deve ser sinalizado de forma explícita para todos os blocos.

Já o modo *skip* é tratado como um caso especial do modo *merge*, e é capaz de atingir uma elevada taxa de compressão em casos onde o erro de predição é muito baixo. Isso se deve porque, ao codificar uma PU com o modo *skip*, o erro de predição é descartado, isto é, a PU é tratada como se o erro de predição fosse zero (SZE; BUDAGAVI; SULLIVAN, 2014). Além disso, o modo *skip* só está disponível em PUs quadradas, ou seja, PUs  $M \times M$  de acordo com a Figura 12.

Dessa forma, quando uma CU é sinalizada através do modo *skip*, isso implica três condições:

- esta CU contém apenas uma PU (uma PU  $M \times M$  compreende uma CU inteira)
- a informação de movimento foi derivada a partir do modo *merge*
- não existe informação residual da predição

Sendo assim, como a informação de movimento é derivada a partir da lista de candidatos *merge* – que é gerada de forma idêntica no codificador e no decodificador – e não há informação residual, o modo *skip* requer apenas uma *flag* indicando que o modo *skip* foi utilizado, e um índice correspondente ao candidato *merge* empregado na codificação.

Testes experimentais conduzidos em (SZE; BUDAGAVI; SULLIVAN, 2014) indicam que em média, é possível reduzir o tamanho do vídeo codificado em 7% sem degradar a qualidade objetiva ao utilizar-se os modos *merge* e *skip*.

### 3.4 Considerações Finais

Este capítulo apresentou as principais ferramentas utilizadas em padrões de codificação de vídeo modernos e suas implementações no padrão HEVC. Durante a codificação, cada região do vídeo é segmentada através de múltiplas possibilidades de particionamento, e para cada possibilidade de particionamento, as previsões intraquadro e interquadros são avaliadas – salvo algumas exceções. Desta forma, cada conjunto de particionamento e modo de previsão (modo intraquadro, vetor de movimento, modo *merge/skip*, etc) competem através do RDO para definir o melhor método de representação para cada região do quadro. A avaliação de um conjunto tão grande de possibilidades faz com que os codificadores de vídeo atuais apresentem uma elevada complexidade computacional, que se reflete diretamente nos tempos de processamento.

As ferramentas presentes nos padrões de codificação de vídeo convencionais foram desenvolvidas considerando as diferentes características presentes neste tipo de conteúdo, e desta forma, o comportamento do codificador pode variar bastante de acordo com o conteúdo do vídeo sendo codificado visto que as características de movimento e textura do vídeo influenciam diretamente nos particionamentos e modos de previsão mais utilizados.

Visto que o comportamento das ferramentas de codificação está diretamente relacionado ao conteúdo do vídeo sendo codificado, é possível que as distorções causadas pela projeção ERP utilizada em vídeos omnidirecionais cause um comportamento diferenciado nestas ferramentas de codificação, dado que tais distorções não ocorrem em vídeos convencionais.

Sendo assim, fica claro que existem desafios e oportunidades para alcançar a codificação de vídeos omnidirecionais de forma eficiente quando padrões de codificação de vídeos convencionais são utilizados.

## 4 PROPOSTAS DA LITERATURA PARA A CODIFICAÇÃO DE VÍDEOS OMNIDIRECIONAIS

Embora seja possível realizar a codificação de vídeos omnidirecionais utilizando o padrão HEVC ou qualquer outro padrão de codificação de vídeo convencional, estes padrões foram desenvolvidos visando a exploração das características de vídeos convencionais, sendo assim, é possível que eles não explorem da melhor forma possível as características de vídeos omnidirecionais que acabam surgindo na etapa de projeção. Entre as características específicas dos vídeos omnidirecionais, é possível citar o esticamento em regiões específicas do vídeo de acordo com a projeção utilizada, as bordas bem definidas e descontinuidades em projeções geométricas como *cube-map* e octaédrica, a característica esférica do vídeo que cria continuidades entre lados opostos dos quadros, o fato de que o usuário assiste a apenas uma porção do vídeo (*viewport*) de cada vez, e além disso vídeos omnidirecionais demandam resoluções muito maiores do que vídeos convencionais para proporcionarem a mesma experiência para o usuário. Dessa forma, é necessário que sejam feitas adaptações nas ferramentas de codificação de vídeo para que estas passem a explorar de forma mais eficiente as características causadas pela projeção de vídeos omnidirecionais. As seções a seguir apresentam os esforços que estão sendo feitos atualmente em direção à padronização da codificação e distribuição de vídeos omnidirecionais, além de propostas feitas para adequar as ferramentas dos padrões de codificação já existentes para melhorar a eficiência da codificação de vídeos omnidirecionais com a infraestrutura presente atualmente.

### 4.1 Esforços para a Padronização

As organizações de padronização têm feito esforços para normatizar a codificação e distribuição de vídeos omnidirecionais. Estes esforços abordam aspectos como a criação de um tipo de mídia próprio para aplicações omnidirecionais, e a implementação de ferramentas específicas ou otimizadas para a codificação de vídeos omnidirecionais em padrões de codificação de vídeo futuros.

O *Moving Pictures Experts Group* (MPEG) é um grupo de especialistas trabalhando em conjunto para definir padrões de compressão e transmissão de áudio e vídeo. Recentemente o MPEG definiu o *Omnidirectional MediA Format* (OMAF) (MPEG, 2019), o primeiro padrão de mídia voltado à distribuição de aplicações omnidirecionais, incluindo vídeo, imagem, áudio e legendas. O OMAF define normas para o pré-processamento dos dados, codificação, e encapsulamento dos arquivos e segmentos.

O OMAF permite a distribuição de vídeos omnidirecionais utilizando a projeção ERP ou *cubemap*. O OMAF também permite que após a projeção o vídeo sofra manipulações regionais, como rotacionar as faces do cubo, reduzir a resolução de regiões polares em vídeos ERP, entre outros (HANNUKSELA; WANG; HOURUNRANTA, 2019).

Do ponto de vista da codificação, o OMAF possui três perfis que utilizam ou o padrão H.264 AVC (JVT, 2003) ou o HEVC (JCT-VC, 2013) para a codificação do vídeo, e estes perfis se dividem em codificação independente e dependente de *viewport*. Quando a codificação independente de *viewport* é utilizada, todo o vídeo é codificado com a mesma qualidade através do padrão HEVC. Por outro lado, caso a codificação dependente de *viewport* seja utilizada, o OMAF emprega ou o padrão H.264 ou o HEVC. Nesse caso, o vídeo omnidirecional é codificado em vários níveis de qualidade de forma que o *viewport* que o usuário está assistindo é transmitido em alta qualidade, enquanto que o restante do vídeo (fora do campo de visão) é transmitido numa qualidade inferior (HANNUKSELA; WANG; HOURUNRANTA, 2019).

O padrão OMAF tem o objetivo de padronizar as aplicações e distribuição de conteúdo omnidirecional, de forma que diferentes provedores de conteúdo e aplicações sejam interoperáveis. Entretanto, no que tange as ferramentas utilizadas durante a codificação dos vídeos, não são propostas modificações: os padrões H.264 ou HEVC são utilizados de forma direta, apenas com a possibilidade de codificar diferentes regiões do vídeo com qualidades distintas.

Por outro lado, o *Joint Video Experts Team* (JVET) está trabalhando no desenvolvimento de um novo padrão de codificação de vídeo considerando o suporte a vídeos omnidirecionais. O JVET é composto por profissionais de diversas empresas, universidades e institutos de pesquisa com experiência no tema de codificação de vídeo, e o novo padrão sendo desenvolvido é denominado *Versatile Video Coding* (VVC) (JVET, 2019). Para organizar e planejar o progresso do desenvolvimento do VVC em subáreas particulares, o JVET estabeleceu 13 grupos *ad hoc* para trabalhar em temas específicos, sendo que o grupo 13 é responsável por experimentos relacionados a vídeos omnidirecionais (SULLIVAN; OHM, 2018).

Os experimentos realizados por este grupo tem o objetivo de determinar quais abordagens apresentam os melhores resultados durante a codificação de vídeos omnidirecionais, e a partir disto, incorporar as melhores abordagens no padrão de co-

dificação de vídeo VVC. Os experimentos cobrem temas como o uso de diferentes projeções, modificações nas previsões intraquadro e interquadros, otimizações nos filtros do codificador, entre outros.

O documento JVET-K0033 (HANHART; LIN, 2018) apresenta resultados exploratórios sobre a eficiência de codificação de diversas projeções, além do impacto destas projeções no tempo de codificação e decodificação. De modo geral, os resultados demonstram que é possível obter aumentos expressivos na eficiência de codificação com algumas projeções sem causar impactos significativos no tempo de codificação/decodificação, indicando a importância da escolha e padronização da projeção de vídeos omnidirecionais.

O documento JVET-L0033 (HANHART; LIN; PUJARA, 2018) apresenta os resultados de diversas técnicas propostas para reduzir artefatos visuais nas bordas de projeções geométricas como a *cubemap*. Entre estas técnicas é possível citar alterações na previsão intraquadro para que apenas amostras dentro da mesma face sejam usadas durante a previsão, e que a lista de MPMs seja gerada a partir de blocos que são vizinhos tanto na projeção quanto no espaço esférico. Também foram analisadas técnicas que replicam amostras esfericamente vizinhas ao redor das faces da projeção para melhorar o desempenho da projeção interquadros. Neste mesmo documento também são avaliadas modificações nos filtros durante a codificação.

O documento JVET-M0033 (HANHART; LIN; PUJARA, 2019) expande as análises anteriores, avaliando outras técnicas para reduzir artefatos visuais em regiões descontínuas do vídeo. Além disso, no documento JVET-L0647 (BOYCE, 2018) os pesquisadores levantam outras questões pertinentes à codificação de vídeos omnidirecionais, como a alteração das estruturas de particionamento para que os particionamentos coincidam com as regiões descontínuas das projeções, e se o padrão VVC deve possuir um perfil de codificação específico para vídeos omnidirecionais ou se estas novas ferramentas devem estar presentes no perfil de codificação principal.

Os dados recentemente apresentados demonstram que o padrão OMAF (MPEG, 2019) é um forte candidato à padronização na distribuição de vídeos omnidirecionais, entretanto este emprega padrões de codificação de vídeo já definidos, como o H.264 (JVT, 2003) e HEVC (JCT-VC, 2013). Por outro lado, o futuro padrão VVC (JVET, 2019) já está sendo desenvolvido visando o suporte a vídeos omnidirecionais, entretanto, este só estará disponível no futuro. Sendo assim, com a atual demanda por vídeos omnidirecionais a comunidade tem proposto adaptações e otimizações para as ferramentas presentes nos padrões de codificação atuais para que possam ser empregadas diretamente na infraestrutura já existente, com o auxílio ou não do OMAF.

## 4.2 Adaptações na Predição Intraquadro

Como as projeções utilizadas para adequar os vídeos omnidirecionais aos padrões de codificação de vídeo convencionais criam distorções geométricas na imagem, é natural que a predição intraquadro apresente um comportamento distinto durante a codificação de vídeos convencionais e omnidirecionais. Dessa forma, alguns trabalhos foram propostos na literatura para otimizar ou analisar de forma mais adequada o comportamento da predição intraquadro quando aplicada a vídeos omnidirecionais projetados.

Como vídeos omnidirecionais apresentam uma resolução significativamente maior do que vídeos convencionais para que os usuários tenham uma mesma percepção de qualidade, a codificação de vídeos omnidirecionais costumam demandar mais tempo do que a codificação de vídeos convencionais. Dessa forma, os autores de (WANG et al., 2017) propõem um esquema para reduzir o tempo de processamento da predição intraquadro sobre vídeos omnidirecionais. O esquema proposto em (WANG et al., 2017) trabalha sobre a projeção ERP e é composto por diversas técnicas, que podem ser agrupadas em dois conjuntos: técnicas genéricas e técnicas específicas a vídeos omnidirecionais.

No que diz respeito as técnicas genéricas, é possível citar uma alteração na derivação dos MPMs (*Most Probable Modes*), reduzir o número de modos avaliados durante o MPM e descartar a avaliação de alguns tamanhos de bloco pouco prováveis. Durante a derivação dos MPMs, quatro PUs vizinhas são avaliadas para derivar modos mais prováveis – duas a mais do que o codificador de referência. Além disso, a derivação dos MPMs ocorre antes do cálculo do RMD. Durante o cálculo do RMD, ao invés de avaliar todos 35 modos de predição intraquadro, a técnica proposta consiste em avaliar um subconjunto dos modos de predição, de forma que para PUs maiores, menos modos de predição são avaliados. Após definir qual dos modos de predição obtém o melhor desempenho, os vizinhos deste modo são avaliados, e depois os vizinhos destes, e assim os próximos vizinhos são avaliados até que não hajam vizinhos não-avaliados. Por fim, os modos avaliados durante o RMD que obtiveram os melhores resultados são enviados para o RDO.

Além disso, os autores compararam as dimensões e a complexidade de textura da PU sendo codificada com um conjunto de PUs vizinhas. Caso a CU atual tenha dimensões maiores do que todas as PUs vizinhas e a textura da PU atual seja mais complexa do que de todas as PUs vizinhas, então o cálculo de RDO é descartado e a PU atual é subdividida. Por outro lado, caso a PU atual apresente dimensões menores do que todas as PUs vizinhas e a sua textura seja a menos complexa entre as PUs vizinhas, então esta PU não é mais subdividida.

Já em relação as técnicas que exploram as características específicas dos vídeos

omnidirecionais, a técnica de (WANG et al., 2017) consiste somente em reduzir de forma mais agressiva o número de modos avaliados durante o RMD nas regiões onde a projeção ERP causa maiores distorções.

O trabalho (STORCH, 2018) realiza uma análise sobre o comportamento da predição intraquadro aplicada a vídeos omnidirecionais na projeção ERP no que diz respeito às escolhas de modos de predição. Durante esta análise, os autores constataram que na região central de vídeos omnidirecionais todos os modos de predição angulares ocorrem com uma frequência semelhante, de forma muito similar ao que ocorre nos quadros de vídeos convencionais.

Por outro lado, esta mesma análise para as regiões próximas ao limite superior e inferior do quadro revela que a predição intraquadro tende a escolher modos de predição horizontalmente orientados, isto é, próximos ao modo 10 (ver Figura 13), com uma frequência muito maior do que demais inclinações. Essa tendência é causada pelo esticamento horizontal causado pela projeção ERP, que por sua vez aumenta a ocorrência de texturas horizontais.

A partir destas análises, uma proposta de redução de complexidade é proposta em (STORCH, 2018) e (STORCH et al., 2019). Esta técnica considera que cada quadro é composto por regiões de distorção distinta, e durante a codificação de PUs em cada uma destas regiões a técnica proposta avalia três aspectos: a distribuição na taxa de ocorrência dos modos de predição de acordo com a região sendo codificada (gerada a partir da análise prévia), quais modos obtiveram melhor desempenho durante o RMD, e quais modos foram derivados como MPMs. Com estes três fatores, a técnica proposta reduz a complexidade da codificação de vídeos omnidirecionais através da redução do número de modos que é enviado para a avaliação RDO.

Os autores de (ZHANG et al., 2019) propõem uma técnica baseada em *mean of the absolute difference* (MAD) para reduzir o número de tamanhos de bloco e modos de predição avaliados. O MAD é uma métrica que mede a complexidade de textura de uma imagem, e os autores calculam o MAD horizontal e vertical de cada bloco do vídeo, dessa forma, medindo a complexidade da textura no sentido horizontal e vertical, respectivamente.

A parte da técnica responsável por descartar a avaliação de alguns tamanhos de bloco é baseada apenas no MAD horizontal. Antes de avaliar os modos de predição intraquadro para o bloco sendo codificado, o MAD horizontal é calculado. Caso o valor do MAD esteja abaixo de um determinado valor pré-definido, é considerado que o bloco apresenta uma textura muito homogênea. Dessa forma, os modos de predição intraquadro são avaliados para o bloco atual e a avaliação de sub-particionamentos é interrompida, reduzindo a profundidade da árvore de CUs avaliada. Por outro lado, caso o MAD horizontal apresente um valor superior a outro valor pré-definido, é considerado que o bloco apresenta uma textura muito complexa. Nesse caso, nenhum

modo de predição intraquadro é avaliado para o bloco atual e este é subdividido em blocos menores imediatamente. Por fim, caso o valor do MAD horizontal esteja entre  $\alpha$  e  $\beta$ , a codificação é realizada normalmente.

A parte da técnica responsável por reduzir o número de modos de predição avaliados utiliza tanto o MAD horizontal quanto o vertical. Caso o MAD vertical seja maior do que o MAD horizontal, então o bloco possui texturas mais orientadas na horizontal e apenas os modos orientados na horizontal serão avaliados (17 dos 33 modos angulares) durante o RMD. Caso contrário, apenas os modos orientados na vertical serão avaliados (17 dos 33 modos angulares, sendo o modo 18 comum para os dois conjuntos) durante o RMD. Entretanto, ao invés de avaliar todos os 17 modos angulares, os autores avaliam os modos de forma sub-amostrada para descartar a avaliação de modos pouco prováveis. Por fim, o RMD envia apenas os dois modos com melhor desempenho para o RDO, ao invés dos três ou oito modos padrões do HM (JCT-VC, 2017).

### 4.3 Adaptações na Predição Interquadros

A projeção utilizada para representar vídeos omnidirecionais de forma planar também causa algum impacto na predição interquadros e na estimativa de movimento. Codificadores de vídeo convencionais assumem que a movimentação dos objetos no vídeo é puramente translacional, e com isso o movimento pode ser caracterizado por um vetor de movimento translacional. Entretanto, movimentos translacionais no vídeo omnidirecional podem ser mapeados de forma distorcida para a projeção, além de que blocos quadrados no vídeo projetado possuem formas arbitrárias na superfície esférica. Essa característica faz com que a predição interquadros presente nos padrões de codificação de vídeo convencionais não explorem de forma adequada a movimentação presente em vídeos omnidirecionais, e alguns trabalhos foram propostos para aumentar seu desempenho. Além disso, como vídeos omnidirecionais apresentam resoluções significativamente maiores do que vídeos convencionais, o tempo de codificação destes vídeos também tende a ser maior. Sendo assim, também existem trabalhos propondo esquemas de redução de complexidade para a predição interquadros de vídeos omnidirecionais.

No trabalho (DESIMONE et al., 2017) os autores propõem a ideia de *plano de movimento* para realizar a estimativa de movimento. Os autores propõem que um movimento arbitrário na superfície esférica pode ser mapeado num movimento translacional no plano de movimento, cujo vetor normal é perpendicular a superfície da esfera. Sendo assim, durante a predição de um determinado bloco é gerado um plano de movimento cuja normal incide de forma perpendicular ao bloco que deve ser predito na superfície da esfera. Esse plano de movimento possui dimensões reduzidas para

que possa mapear uma determinada parte da superfície esférica sem causar distorções significativas. Uma vez gerado o plano de movimento, um BMA (*Block Matching Algorithm*) convencional pode ser utilizado para encontrar o bloco mais similar dentro do plano de movimento. Esta técnica é capaz de modelar o movimento de vídeos omnidirecionais de forma mais adequada que o movimento puramente translacional de codificadores convencionais, sendo assim, é possível atingir uma maior taxa de compressão para uma mesma qualidade visual.

Já no trabalho (VISHWANATH; NANJUNDASWAMY; ROSE, 2017) os autores vão além e propõem realizar a estimativa de movimento diretamente no domínio esférico. Para isso, sempre que uma PU do vídeo projetado deve ser predita, esta PU é mapeada novamente na superfície esférica do vídeo omnidirecional. Em seguida, uma área de busca radial ao redor do bloco a ser predito é definido. Essa área de busca possui pontos de amostragem – onde os blocos serão avaliados a procura da maior similaridade – mais densos no centro da área de busca, e mais esparsos conforme se afastam do centro da área de busca. Essa área de busca prioriza vetores de movimento com magnitudes menores, visto que estes são mais comuns. Além disso, uma área de busca circular avalia os blocos na superfície de uma esfera de forma mais adequada do que uma área de busca retangular. Em seguida os blocos dentro da área de busca são avaliados para encontrar o bloco mais similar ao bloco a ser predito, e este bloco é caracterizado por um vetor de movimento rotacional que permite deslocar o bloco a ser predito até o bloco com maior similaridade.

Os autores de (RAY; JUNG; LARABI, 2018) propõem um esquema de redução de complexidade para a predição interquadros de vídeos omnidirecionais na projeção ERP. Esse esquema altera o comportamento da estimativa de movimento fracionária (FME) e o tamanho máximo de bloco baseado em dois conjuntos de limiares. Como as regiões polares de vídeos omnidirecionais possuem alta redundância horizontal, os autores defendem que a FME no sentido horizontal pode ser desnecessária nestas regiões. Sendo assim, na região central do vídeo a FME é feita de forma normal, nas regiões mais próximas dos polos a FME é feita apenas no sentido vertical (estimativa de movimento horizontal com precisão inteira), e nas regiões intermediárias a FME no sentido horizontal é feita apenas com precisão de meio pixel, enquanto que no sentido vertical é feita tanto com um meio quanto um quarto de pixel.

Em relação aos tamanhos de bloco, os autores utilizaram uma versão do codificador que permite que cada CU folha da *quadtree* possa ser subdividida novamente através de uma *binary tree*. Com esse método de particionamento, eles dividem o vídeo em quatro faixas de latitude. CUs em latitudes menores, próximo ao centro do quadro, podem ter quaisquer dimensões permitidas pelo padrão. Conforme CUs mais próximas dos polos são avaliadas, em outras faixas de latitude, o tamanho mínimo de CU é aumentado para que a árvore de CUs possua uma profundidade menor. Ao uti-

lizar estas duas técnicas, os autores conseguem simplificar o codificador nas regiões mais redundantes e obter uma redução de complexidade. Entretanto, o codificador empregado nestes testes foi o *Joint Exploration Model*, que é um codificador criado de forma temporária e com caráter exploratório para avaliar formas de aumentar a eficiência de codificação do padrão HEVC. Dessa forma, é possível que os resultados obtidos pelos autores não estejam de acordo com o que aconteceria num padrão de codificação definido.

Os autores de (LIU; SONG; ZHANG, 2019) utilizam um classificador *k-nearest neighbors* (KNN) para limitar a profundidade da árvore de CUs avaliada durante a predição interquadros. Para isso, os autores utilizam como parâmetros para o classificador a posição vertical do bloco dentro do quadro, visto que o esticamento da projeção ERP é função de tal posição, e também utilizam o número de bits necessário para codificar a CU atual no melhor caso.

Para utilizar esta técnica, os autores agrupam os quadros do vídeo em quadros de treino e quadros de predição, e estes grupos se alternam no decorrer do vídeo. Nos quadros do grupo de treino, a codificação é realizada sem interferência e os dados da codificação são salvos para uso futuro. Durante a codificação dos quadros no grupo de predição, a posição e número de bits necessário para codificar o bloco atual é enviado para o classificador KNN. O classificador compara estes dados com os dados obtidos da codificação dos quadros do grupo de treino e determina se o bloco atual precisa ser subdividido em blocos menores ou se a avaliação de blocos menores pode ser descartada.

#### 4.4 Adaptações para a Transmissão

Embora vídeos omnidirecionais compreendam toda uma superfície esférica ao redor da câmera e, consequentemente, do usuário, os usuários assistem a apenas um *viewport* deste vídeo por vez. Sendo assim, transmitir um vídeo completo em qualidade máxima desde o servidor até o usuário final caracteriza um desperdício de banda, visto que uma largura de banda considerável é utilizada para transmitir regiões do vídeo que não são assistidas pelo usuário. Dado este cenário, alguns trabalhos foram propostos para reduzir o consumo de banda na transmissão de vídeos omnidirecionais explorando o fato de que apenas uma porção do vídeo é assistido por vez.

Em (ZARE et al., 2016) os autores abordam esse problema utilizando *Tiles* e codificando o vídeo em duas resoluções distintas. *Tiles* compreendem uma ferramenta do padrão HEVC que divide cada quadro do vídeo em arranjos retangulares, que podem ser codificados e decodificados de forma independente (SULLIVAN et al., 2012). Com esta ferramenta, os autores dividem o vídeo em arranjos de *Tiles*  $6 \times 3$ ,  $4 \times 3$  e  $4 \times 2$ , ou seja, todos os arranjos possuem mais colunas de *Tiles* do que linhas.

Com esta divisão, todos os *Tiles* são codificados duas vezes e armazenados no servidor. Uma destas codificações é feita na resolução original do vídeo enquanto que a outra é feita numa resolução menor, que demanda menor largura de banda para ser transmitida. Enquanto o usuário assiste a um vídeo nesse sistema, o dispositivo de reprodução envia uma mensagem ao servidor indicando o *viewport* que está sendo assistido. Quando o servidor recebe esta informação, ele envia os *Tiles* que correspondem ao *viewport* atual em alta resolução, enquanto que os *Tiles* que correspondem a regiões fora do *viewport* são enviados em baixa resolução. Com este método, é possível reduzir a quantidade de dados que é enviada ao usuário.

Dessa forma, o usuário é capaz de assistir ao *viewport* desejado com a qualidade máxima. Caso o usuário mude ligeiramente o *viewport*, é provável que o novo *viewport* ainda seja composto pelo mesmo conjunto de *Tiles*, sendo assim, o usuário não tem sua experiência prejudicada. Entretanto, caso o usuário mude o *viewport* de maneira brusca ele vai assistir ao vídeo em baixa resolução por alguns instantes até que o servidor detecte a variação de *viewport* e envie um conjunto de *Tiles* de alta resolução diferente.

Já os autores de (HOOFT et al., 2019) propõem uma técnica que abrange predição de *viewport*, adaptação de qualidade de acordo com o *viewport* atual, e melhorias no servidor *Hypertext Transfer Protocol* (HTTP). Para isso, o vídeo é dividido em *Tiles* e cada *Tile* é codificado em várias qualidades. Para realizar a predição de *viewport*, os autores consideram que o movimento (que leva à variação do *viewport*) comprehende uma trajetória na superfície da esfera. Sendo assim, quando o usuário realiza uma determinada trajetória num período de tempo, esta trajetória é estendida unidirecionalmente para prever a trajetória futura. Entretanto, visto que a movimentação do usuário é altamente volátil, apenas uma fração da trajetória é estendida para reduzir o erro de predição.

Os autores utilizam *Tiles* para adaptar a qualidade do vídeo à capacidade da rede, sendo que a técnica é escalável para um número arbitrário de *Tiles*. Durante a execução do vídeo, o cliente percebe a largura de banda disponível. Em seguida, os *Tiles* são divididos em dois grupos: aqueles cujo centro está dentro do *viewport* atual, e aqueles cujo centro está fora. Após isso, é sinalizado que todos os *Tiles* devem ser baixados na menor qualidade, e em seguida o algoritmo de alocação de banda entra em funcionamento.

O algoritmo aumenta a qualidade dos *Tiles* dentro do *viewport* de forma homogênea até que ou toda a banda seja consumida ou todos os *Tiles* dentro do *viewport* estejam na qualidade máxima. Caso seja possível baixar todos os *Tiles* dentro do *viewport* com a qualidade máxima, uma técnica semelhante é empregada para aumentar a qualidade dos *Tiles* fora do *viewport*. Por fim, os autores propõem o uso de servidores HTTP 2.0 para realizar a transmissão do vídeo. Como o vídeo é segmen-

tado em *Tiles* independentes, o uso de um servidor HTTP convencional faz com que o cliente precise fazer várias requisições para o servidor, uma para cada *Tile*, o que cria uma latência principalmente no início da transmissão. Com o uso do servidor HTTP 2.0, os autores defendem que é possível utilizar o serviço de SERVER PUSH para que apenas uma requisição por parte do cliente acarrete no servidor realizar o PUSH de todos os *Tiles*, reduzindo assim a latência.

## 4.5 Considerações Finais

Este capítulo apresentou as principais características do padrão OMAF, que busca suprir a demanda emergencial por uma padronização na distribuição de vídeos omnidirecionais, além de destacar alguns aspectos que o JVET vem discutindo para que o futuro padrão de codificação VVC tenha ferramentas dedicadas à codificação de vídeos omnidirecionais. Entretanto, o OMAF prevê o uso de padrões de codificação de vídeo convencionais na sua estrutura, e o padrão VVC ainda está sob desenvolvimento, então nenhum deles é capaz de otimizar a atual demanda por codificação eficiente de vídeos omnidirecionais.

Além disso, foram apresentados trabalhos propondo otimizações na codificação intraquadro, interquadros, e na transmissão de vídeos omnidirecionais considerando os padrões de codificação atuais. Os trabalhos apresentados propõem técnicas para reduzir a complexidade da predição intraquadro e interquadros, aumentar a eficiência de codificação atingida pela predição interquadros, além de reduzir a largura de banda necessária para transmitir vídeos omnidirecionais. Entretanto, ainda existem desafios e oportunidades de pesquisa que não foram exploradas pelos trabalhos da literatura.

Embora o trabalho (WANG et al., 2017) utilize um sistema para não avaliar alguns tamanhos de bloco, a técnica utilizada é agnóstica à região do vídeo sendo codificada, isto é, ela não considera nenhuma característica da projeção e poderia apresentar os mesmos resultados em vídeos convencionais. Sendo assim, faz-se necessário um estudo acerca de como os tamanhos de bloco são escolhidos durante a predição intraquadro de vídeos omnidirecionais que dê subsídio para uma técnica de redução de complexidade especializada para vídeos omnidirecionais.

Já o trabalho (RAY; JUNG; LARABI, 2018) explora a simplificação da FME e alteração nos tamanhos de bloco permitidos para diferentes regiões do vídeo, entretanto, os resultados foram obtidos num codificador experimental que não reflete a realidade de padrões de codificação definidos. Dessa forma, faz-se necessário avaliar técnicas de redução de complexidade baseadas na alteração da precisão da FME num codificador padronizado.

Além dos aspectos já mencionados, não existem trabalhos que avaliem a correlação entre as redundâncias em regiões distorcidas pela projeção ERP e a utilização do

modo de predição *skip*. Este modo de predição costuma ser usado em regiões homogêneas onde vários blocos compartilham informação de movimento, sendo assim, é possível definir técnicas de escolha rápida de modo *skip* que ignorem a estimativa de movimento convencional em regiões distorcidas, levando a elevadas reduções de complexidade da predição interquadros.

Por fim, como a projeção ERP de vídeos omnidirecionais altera as texturas do vídeo e a trajetória dos movimentos, é possível explorar as características da projeção e seus impactos na escolha dos vetores de movimento para reduzir a complexidade da estimativa de movimento.

## 5 AVALIAÇÃO DO COMPORTAMENTO DA CODIFICAÇÃO DE VÍDEOS OMNIDIRECIONAIS

Com base nas ferramentas de codificação de vídeo presentes nos padrões atuais e descritas no Capítulo 3 e nos desafios e oportunidades de pesquisa mencionados no final do Capítulo 4, este capítulo busca realizar um conjunto de análises para identificar o comportamento das ferramentas de codificação para vídeos omnidirecionais na projeção ERP quando comparados a vídeos convencionais, para que este comportamento possa ser explorado por técnicas de redução de complexidade para codificação destes vídeos omnidirecionais. Todas as análises conduzidas neste capítulo são baseadas na seguinte hipótese geral: o comportamento das ferramentas de codificação de vídeo é diferente para vídeos omnidirecionais gerados pela projeção ERP quando comparado com vídeos convencionais. Com base nesta hipótese geral, quatro hipóteses específicas para ferramentas de codificação foram avaliadas:

- **Hipótese 1** O esticamento horizontal causado pela projeção ERP aumenta a homogeneidade dos blocos e torna-os mais propícios a serem codificados com o modo de predição *skip*
- **Hipótese 2** A projeção ERP estica as texturas da superfície esférica e isso altera a magnitude e orientação dos vetores de movimento nas regiões distorcidas
- **Hipótese 3** A projeção ERP gera amostras em posições fracionárias na superfície esférica para mapeá-los na projeção retangular, fazendo com que a estimativa de movimento fracionária no sentido horizontal gere amostras redundantes e cause pequenos ganhos em eficiência de codificação
- **Hipótese 4** O esticamento horizontal causado pela projeção ERP aumenta a homogeneidade nas regiões distorcidas do vídeo, tornando tais regiões mais propícias a serem codificadas com blocos maiores durante a predição intraquadro

## 5.1 Metodologia de Avaliação

As mais recentes Condições Comuns de Teste para a Codificação de Vídeos Omnidirecionais (CTCs360) (BOYCE et al., 2017) de acordo com o padrão HEVC foram utilizadas para realizar as avaliações descritas nas próximas Seções. Este documento descreve quais configurações devem ser empregadas durante a codificação de vídeos omnidirecionais para que os resultados sejam comparáveis com os demais trabalhos da literatura. Além disso, as Condições Comuns de Teste para a Codificação de Vídeos Convencionais (CTCs) (SHARMAN; SUEHRING, 2017) foram utilizadas para codificar vídeos convencionais.

Todas as codificações foram realizadas com o *software* de referência do padrão HEVC na versão 16.16 (HM-16.16) (JCT-VC, 2017). Durante a codificação de vídeos omnidirecionais, a biblioteca 360Lib na versão 5.0 (360Lib-5.0) (YE; ALSHINA; BOYCE, 2017) foi utilizada em conjunto com o HM-16.16.

Seguindo as recomendações das CTCs convencionais e omnidirecionais, os valores de QP 22, 27, 32 e 37 foram utilizados, tanto para vídeos convencionais quanto omnidirecionais.

Nos casos onde ferramentas de predição interquadros são avaliadas a configuração de codificação *Random Access* é utilizado, enquanto que as avaliações da predição intraquadro são realizadas utilizando-se a configuração *All Intra*.

### 5.1.1 Sequências Avaliadas e Respectivas Resoluções

Entre outros aspectos, as CTCs360 definem quais sequências de vídeo devem ser utilizadas durante a codificação. No total as CTCs360 disponibilizam doze sequências padronizadas, de resoluções  $3840 \times 1920$  (4k),  $6144 \times 3072$  (6k) e  $8192 \times 4096$  (8k). Entretanto, apenas quatro destas sequências foram utilizadas durante as avaliações para não enviesar possíveis técnicas provenientes destas análises, e seus nomes e respectivas resoluções são apresentados nas colunas *Sequência* e *Resolução original* da Tabela 1.

Além disso, as CTCs360 preveem que sequências originalmente 4k devam ser convertidas para a resolução  $3328 \times 1664$  antes da codificação, enquanto que sequências originalmente 6k e 8k devam ser convertidas para a resolução  $4096 \times 2048$  antes da

Tabela 1 – Vídeos omnidirecionais utilizados durante as avaliações e respectivas resoluções

Vídeo	Resolução original	Resolução de avaliação
<i>AerialCity</i>	$3840 \times 1920$	$3328 \times 1664$
<i>PoleVault</i>	$3840 \times 1920$	$3328 \times 1664$
<i>Broadway</i>	$6144 \times 3072$	$3328 \times 1664$
<i>SkateboardInlot</i>	$8192 \times 4096$	$3328 \times 1664$

codificação (BOYCE et al., 2017). Entretanto, algumas avaliações realizadas buscam compreender o comportamento espacial das ferramentas de codificação, e a avaliação de sequências codificadas em múltiplas resoluções pode dificultar a comparação e agrupamento destes dados. Sendo assim, durante as avaliações as sequências *Broadway* e *SkateboardInLot* foram codificadas na resolução  $3328 \times 1664$ , assim como as sequências 4k, para que as avaliações espaciais mantenham as mesmas dimensões, como apresentado na coluna *Resolução de avaliação* da Tabela 1. Todas estas mudanças de resolução foram realizadas através da biblioteca 360Lib (YE; ALSHINA; BOYCE, 2017).

Por fim, em todas as avaliações o comportamento da codificação de vídeos omnidirecionais na projeção ERP foi comparado com a codificação de vídeos convencionais para detectar as diferenças na codificação dos dois tipos de conteúdos.

As sequências de teste convencionais utilizadas durante estas avaliações foram extraídas das CTCs convencionais (SHARMAN; SUEHRING, 2017) e são apresentadas na Tabela 2. De acordo com a Tabela 2, durante a avaliação da predição intraquadro as sequências de vídeo convencionais de resolução  $1920 \times 1080$  *BasketballDrive*, *BQTerrace*, *Cactus*, *Kimono* e *ParkScene* foram utilizadas. Já durante as avaliações do comportamento da predição interquadros, as sequências de vídeo convencionais de resolução  $2560 \times 1600$  *NebutaFestival*, *PeopleOnStreet*, *SteamLocomotiveTrain* e *Traffic* foram utilizadas. O motivo da escolha destas sequências de vídeo convencionais se dá pela similaridade de resolução em relação aos vídeos omnidirecionais: durante as avaliações da predição intraquadro – no início deste trabalho – os vídeos de maior resolução disponíveis possuíam resolução  $1920 \times 1080$ , e no decorrer das avaliações foram obtidas sequências de resolução  $2560 \times 1600$ , que foram utilizadas nas avaliações da predição interquadros posteriores.

Tabela 2 – Vídeos convencionais utilizados durante as avaliações

Vídeo	Resolução	Predição avaliada
<i>BasketballDrive</i>	$1920 \times 1080$	intraquadro
<i>BQTerrace</i>	$1920 \times 1080$	intraquadro
<i>Cactus</i>	$1920 \times 1080$	intraquadro
<i>Kimono</i>	$1920 \times 1080$	intraquadro
<i>ParkScene</i>	$1920 \times 1080$	intraquadro
<i>NebutaFestival</i>	$2560 \times 1600$	interquadros
<i>PeopleOnStreet</i>	$2560 \times 1600$	interquadros
<i>SteamLocomotiveTrain</i>	$2560 \times 1600$	interquadros
<i>Traffic</i>	$2560 \times 1600$	interquadros

### 5.1.2 Avaliação de Qualidade

A avaliação de qualidade de vídeos omnidirecionais por si só já apresenta desafios. Vídeos omnidirecionais na projeção ERP apresentam um esticamento variável de acordo com a posição vertical das amostras dentro do quadro, de modo que na região central do quadro o esticamento é quase nulo enquanto que nas regiões próximas às extremidades superior e inferior ele é muito severo. A partir disso, um conjunto de amostras no vídeo projetado ERP pode corresponder a um conjunto muito menor de amostras na superfície do vídeo esférico. Dessa forma, fica visível que a qualidade de vídeos omnidirecionais na projeção ERP não pode ser avaliada da mesma forma que vídeos convencionais visto que as diferentes amostras do vídeo projetado apresentam impactos distintos na qualidade do vídeo esférico.

A qualidade de vídeos convencionais costuma ser avaliada utilizando-se o *Peak Signal-to-Noise Ratio* (PSNR), que mede a quantidade de ruído inserido no vídeo pelo processo de codificação. A partir disso, diferentes variações do PSNR foram desenvolvidas para avaliar a qualidade de vídeos omnidirecionais, e o *Weighted-to-Spherically Uniform PSNR* (WS-PSNR) é uma destas variações que apresenta os melhores resultados (YU; LAKSHMAN; GIROD, 2015). Quando o WS-PSNR é utilizado, a qualidade do vídeo projetado é avaliada através do PSNR, entretanto, o ruído em cada posição dentro do quadro é ponderado de acordo com o fator de esticamento da superfície esférica para a projeção em questão (YU; LAKSHMAN; GIROD, 2015). Dessa forma, quando a projeção ERP é utilizada, as amostras na região do equador que apresentam esticamento suave ou até mesmo nulo, tem um impacto mais significativo no cálculo da qualidade global, enquanto que as amostras nas regiões superior e inferior, onde o esticamento é mais severo, tem um impacto menos significativo.

Embora o WS-PSNR seja capaz de avaliar a qualidade visual do vídeo codificado, é necessário que a qualidade do vídeo seja relacionada de alguma forma com a taxa de compressão obtida, visto que o objetivo da codificação de vídeo é alcançar a maior taxa de compressão possível com a menor degradação visual possível. Dessa forma, para avaliar a eficiência de codificação, isto é, a taxa de compressão junto da qualidade visual, utiliza-se o *Bjontegaard Delta Bitrate* (BD-BR) (BJØNTEGAARD, 2001). O BD-BR é uma métrica que realiza a comparação entre dois codificadores, um codificador alvo e um codificador referência. A comparação é realizada avaliando o aumento de *bitrate* necessário para que o codificador alvo atinja a mesma qualidade visual do codificador referência. Dessa forma, valores positivos de BD-BR indicam que o codificador alvo apresenta uma eficiência de codificação menor que o codificador referência, enquanto que valores de BD-BR negativos indicam que o codificador alvo apresenta uma eficiência de codificação maior. Além disso, o cálculo do BD-BR recebe como entradas o *bitrate* necessário para representar o vídeo codificado e a qualidade do vídeo em PSNR. Entretanto, como discutido anteriormente o PSNR sozi-

nho não é suficiente para medir a qualidade de vídeos omnidirecionais, e o WS-PSNR é uma melhor alternativa visto que é uma variação do PSNR adaptada a vídeos omnidirecionais. Dessa forma, o cálculo de BD-BR de vídeos omnidirecionais é calculado utilizando-se o *bitrate* e o WS-PSNR destes vídeos, como sugerido pelas CTCs360 (BOYCE et al., 2017).

### 5.1.3 Avaliações Baseadas em Bandas

As avaliações conduzidas neste capítulo buscam identificar como as distorções causadas pela projeção ERP interferem no comportamento das ferramentas de codificação. Em alguns casos estas avaliações se dão de forma espacial, e neste caso, o termo *região central* diz respeito às amostras situadas ao centro do quadro, enquanto que *regiões polares* representam amostras que estão situadas arbitrariamente próximas dos extremos superior e inferior do quadro, sem haver um ponto específico que dite a fronteira entre a *região central* e as *regiões polares*.

Entretanto, em algumas avaliações se faz necessário um ponto específico que separe as regiões com maior e menor distorção. Sendo assim, com o objetivo de relacionar de forma mais objetiva o comportamento da codificação de vídeos omnidirecionais com as diferentes intensidades de esticamento causadas pela projeção ERP, algumas avaliações foram realizadas utilizando-se uma abordagem em bandas. Nesses casos, os vídeos foram divididos ou em três bandas ou em cinco bandas, como na Figura 19.

No caso da abordagem em três bandas, a banda superior compreende 25% das amostras situadas mais ao topo do quadro, enquanto que a banda inferior corresponde 25% das amostras mais abaixo do quadro, e a banda central corresponde as amostras situadas na parte central do quadro, compreendendo 50% das amostras do quadro. As bandas superior e inferior juntas são denominadas *bandas polares* visto que elas estão mais próximas das latitudes  $+90^\circ$  e  $-90^\circ$ , que representam o polo norte e polo sul da esfera, respectivamente.

Já no caso da abordagem em cinco bandas, as bandas superior e inferior compreendem 15% das amostras do quadro cada, situadas na parte mais superior e inferior



Figura 19 – Exemplo do primeiro quadro de *PoleVault* dividido em três e cinco bandas

do quadro. A banda central corresponde a 30% das amostras do quadro, e está situada no centro deste. Por fim, as bandas meio-superior e meio-inferior correspondem a 20% das amostras do quadro cada, e estão situadas entre as bandas polares e a banda central. As bandas meio-superior e meio-inferior juntas são denominadas *bandas meio-polares*, visto que estão situadas entre a banda central e as bandas polares.

Quando estas abordagens em bandas são utilizadas, o vídeo é codificado de forma natural, isto é, os quadros não são recortados e a codificação de todas as bandas é feita da mesma forma – exceto quando dito o contrário. Entretanto, durante a extração dos dados cada banda é tratada de forma individual para que seja possível avaliar se as decisões tomadas durante a codificação das bandas com maior distorção são diferentes das decisões tomadas na codificação das bandas com menor distorção. Embora a utilização de mais bandas torne a modelagem da distorção mais precisa, algumas avaliações utilizaram três bandas pela maior simplicidade na análise dos dados.

## 5.2 Avaliação das Ocorrências de *Skip*

Esta seção busca investigar a **Hipótese 1** deste trabalho, isto é, o esticamento horizontal causado pela projeção ERP aumenta a homogeneidade dos blocos e torna-os mais propícios a serem codificados com o modo de predição *skip*. Para isso, duas avaliações distintas foram realizadas: uma avaliação espacial das ocorrências do modo *skip*, descrita na Seção 5.2.1, e uma análise em bandas das ocorrências do modo *skip* de acordo com a homogeneidade dos blocos, descrita na Seção 5.2.2.

### 5.2.1 Avaliação Espacial das Ocorrências de *Skip*

Antes de realizar as codificações para esta avaliação o HM-16.16 foi modificado para exportar alguns dados referentes às decisões intermediárias da codificação, como se um bloco foi ou não codificado com o modo *skip*, mesmo para aqueles blocos que não foram utilizados na codificação final. Embora esses dados apresentem decisões que não são ótimos globais visto que não foram as decisões finais da codificação, estes ótimos locais possuem grande relevância para este estudo já que representam a decisão ótima local, naquele momento da codificação.

Após extrair as decisões das codificações, os dados foram separados de acordo com os tamanhos de bloco, e em seguida os dados referentes aos mesmos tamanhos de bloco foram analisados para gerar um mapa indicando a taxa de ocorrência do modo *skip* para cada CTU dentro do quadro.

Para realizar esta análise, foram consideradas apenas as CUs codificadas através da predição interquadros visto que o modo *skip* não está disponível durante a predição intraquadro. Dessa forma, foi avaliado com que frequência cada CU  $64 \times 64$ ,  $32 \times 32$ ,

$16 \times 16$  e  $8 \times 8$  foi codificada com o modo *skip* dentro de cada CTU do vídeo. Para gerar a taxa de ocorrência para um determinado tamanho de CU, foi avaliado quantas CUs do respectivo tamanho foram codificados através da predição interquadros para cada CTU, e destas CUs codificadas com a predição interquadros, quantas delas foram codificadas através do modo *skip*.

O resultado destas análises considerando todos os vídeos convencionais simultaneamente – isto é, a média de todos os vídeos – é apresentado na Figura 20. Nesta figura os eixos vertical e horizontal representam a posição da respectiva CTU dentro do quadro, de modo que cada quadrado nestes gráficos representa a taxa de ocorrência de *skip* dentro de uma CTU particular. Além disso, a Figura 20 (a) representa a taxa para CUs  $8 \times 8$ , (b) representa a taxa para CUs  $16 \times 16$ , (c) representa a taxa para CUs  $32 \times 32$ , (d) representa a taxa para CUs  $64 \times 64$  e (e) representa a taxa combinada para todos os tamanhos de CU. Para realizar a combinação dos tamanhos de CU, foi atribuído um peso a cada tamanho de CU, onde CUs  $8 \times 8$  tiveram peso 1, CUs  $16 \times 16$  tiveram peso 4, CUs  $32 \times 32$  tiveram peso 16 e CUs  $64 \times 64$  tiveram peso 64. Além disso, a escala de cores vai desde zero até um, onde um valor zero indica que uma CTU nunca foi codificada com o modo *skip*, enquanto que um valor um indica que uma CTU foi codificada com o modo *skip* 100% das vezes.

Ao analisar-se os dados da Figura 20 é possível tirar conclusões sobre como o modo *skip* é utilizado durante a codificação de vídeos convencionais. Ao analisar a Figura 20 (a) fica visível que o modo *skip* foi utilizado para codificar em torno de 80% das CUs  $8 \times 8$  dentro de cada CTU independente da região do vídeo, visto que embora hajam variações entre as CTUs do vídeo, esta variação é muito pequena quando comparada à magnitude dos valores. Dessa forma, é possível concluir que para CUs  $8 \times 8$  o modo *skip* se comporta de forma semelhante independente da região do quadro sendo codificada.

A análise das Figuras 20 (b) e (c) leva à conclusões semelhantes: de modo geral, pode-se dizer que em torno de 70% das CUs  $16 \times 16$  dentro de cada CTU foram codificadas através do modo *skip*, enquanto que 50% das CUs  $32 \times 32$  dentro de cada CTU foram codificadas com o modo *skip*. Em ambos os casos existem algumas variações na taxa de ocorrência de *skip* quando diferentes regiões do quadro são comparadas, entretanto, estas variações não apresentam um padrão definido e possivelmente estão relacionadas exclusivamente ao conteúdo dos vídeos analisados.

A análise da Figura 20 (d) indica que para CUs  $64 \times 64$  um comportamento semelhante também é observado. De modo geral, a taxa de ocorrência do modo *skip* é muito semelhante para todas as CTUs dentro do quadro – salvo algumas oscilações regionais – e em média, o modo *skip* costuma ser utilizado para codificar em torno de 40% das CUs dentro de cada CTU. A comparação das ocorrências de *skip* para os quatro tamanhos de CU também mostra que o modo *skip* é utilizado com maior

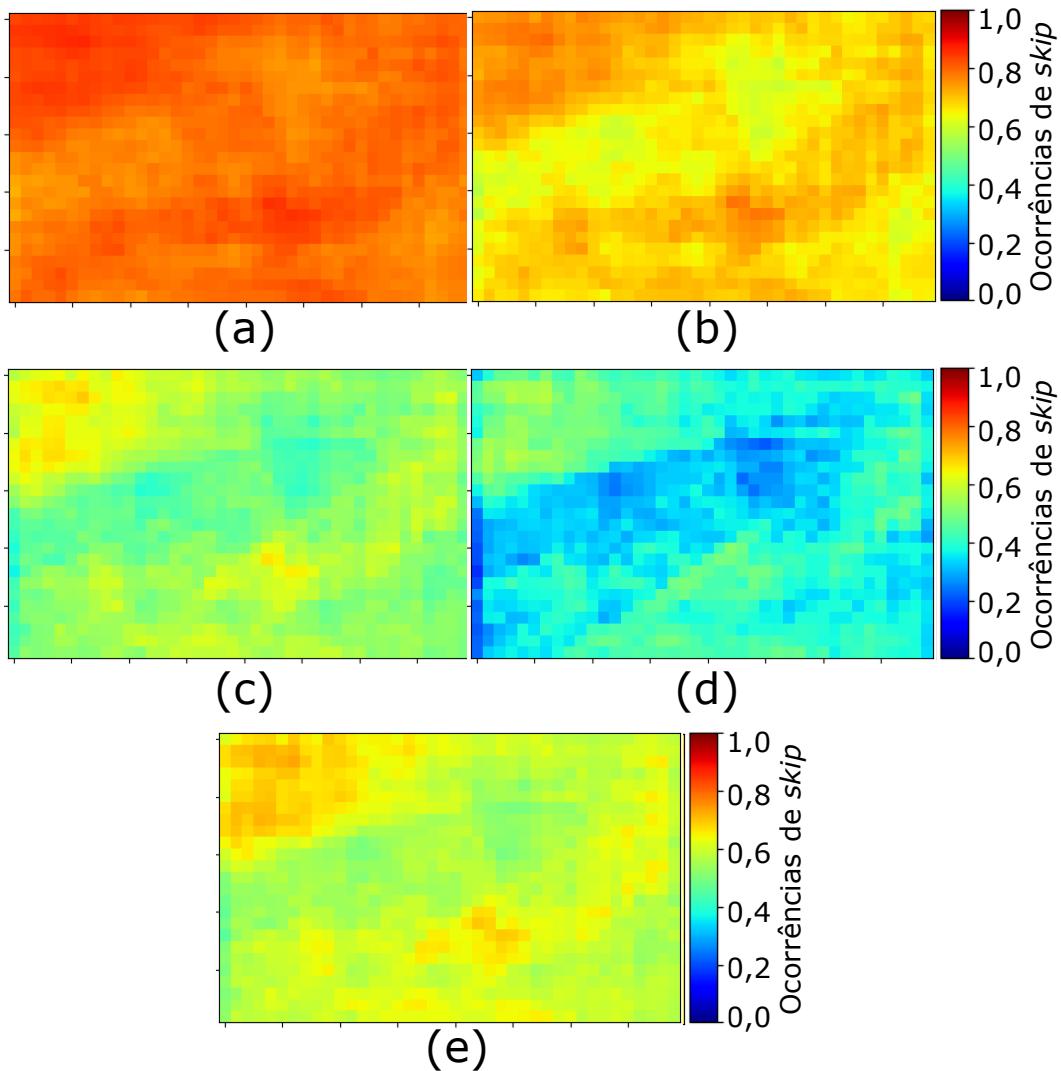


Figura 20 – Taxa de ocorrência do modo *skip* para vídeos convencionais. (a) CUs  $8 \times 8$ , (b) CUs  $16 \times 16$ , (c) CUs  $32 \times 32$ , (d) CUs  $64 \times 64$  e (e) todos tamanhos ponderados

frequência durante a codificação de CUs menores. Por fim, a análise da Figura 20 (e) representa uma visão geral do comportamento do modo *skip* visto que reúne os resultados para todos os tamanhos de CUs. Neste caso, é possível confirmar o comportamento já visto anteriormente: embora existam variações na taxa de ocorrência de modos *skip* ao longo do quadro, estas variações são muito pequenas. Desta forma, pode-se concluir que durante a codificação de vídeos convencionais, não existem regiões do vídeo mais ou menos propícias a serem codificadas com o modo *skip*, isto é, para qualquer que seja a CTU sendo codificada, a probabilidade desta ser codificada através do modo *skip* é sempre muito próxima de um determinado valor relacionado diretamente ao tamanho de CU sendo processado.

Já os resultados de taxa de ocorrência de modos *skip* média para os vídeos omnidirecionais avaliados são apresentados na Figura 21, onde os eixos vertical e horizontal

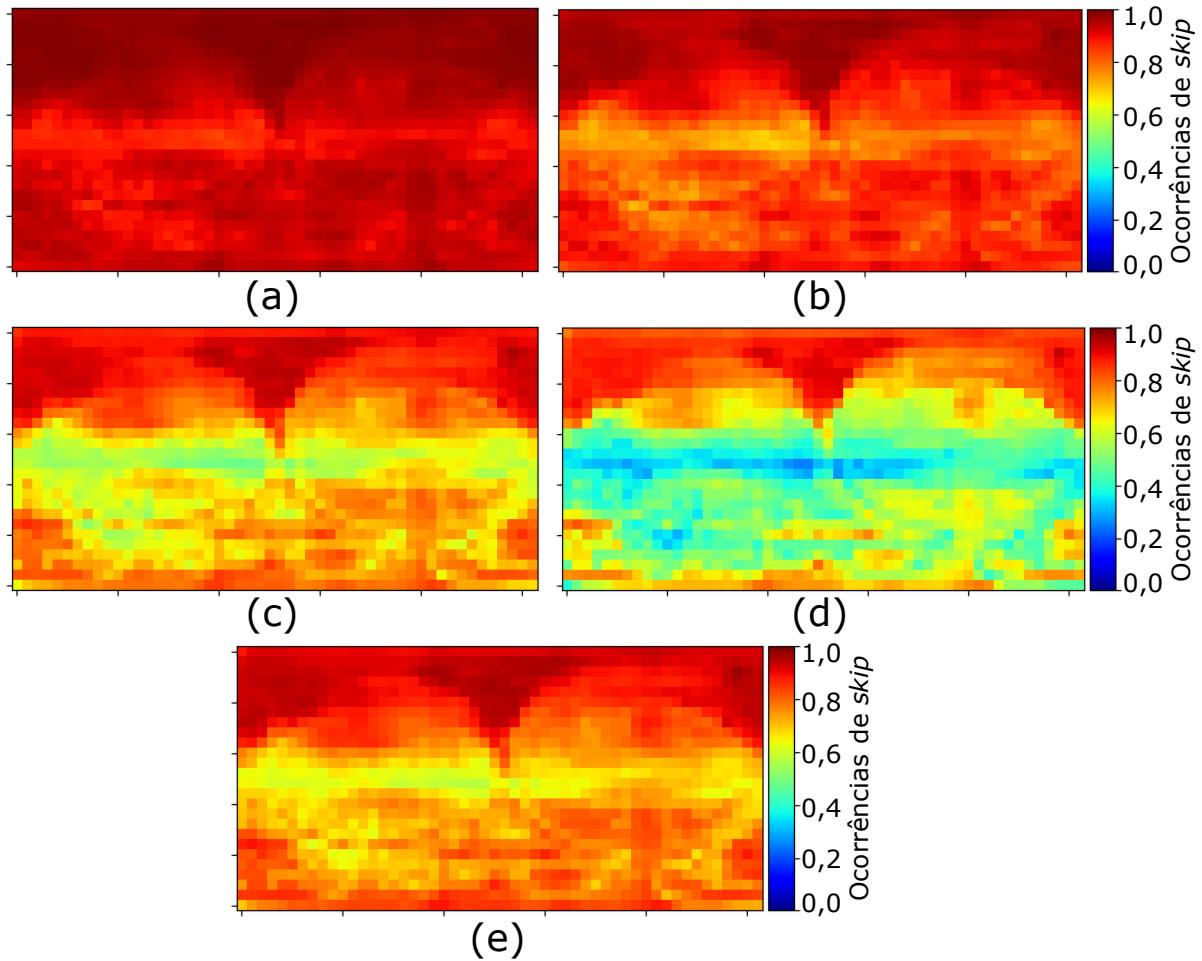


Figura 21 – Taxa de ocorrência do modo *skip* para vídeos omnidirecionais. (a) CUs  $8 \times 8$ , (b) CUs  $16 \times 16$ , (c) CUs  $32 \times 32$ , (d) CUs  $64 \times 64$  e (e) todos tamanhos ponderados

representam a posição de uma respectiva CTU dentro do quadro. A Figura 21 (a) representa a taxa para CUs  $8 \times 8$ , (b) representa a taxa para CUs  $16 \times 16$ , (c) representa a taxa para CUs  $32 \times 32$ , (d) representa a taxa para CUs  $64 \times 64$  e (e) representa a taxa combinada para todos os tamanhos de CU ponderados da mesma forma que para vídeos convencionais.

Quando a Figura 21 é analisada, fica claro que o modo *skip* apresenta um comportamento bastante diferente para vídeos omnidirecionais quando comparado a vídeos convencionais. Para CUs de tamanho  $8 \times 8$ , mais de 90% das CUs dentro de cada CTU são codificadas com o modo *skip*, enquanto que na região superior do vídeo esta taxa quase alcança 100%. De modo geral, a região central do quadro apresenta uma taxa de ocorrência de modos *skip* ligeiramente menor do que no restante do quadro, entretanto, esta diferença é pequena. A análise da Figura 21 (b) mostra um comportamento semelhante para as CUs  $16 \times 16$  de vídeos omnidirecionais, onde a região superior e inferior do quadro apresentam taxas de ocorrência de *skip* visivelmente maiores do

que a região central: enquanto que na região superior atinge-se em média 95% de ocorrência de *skip*, na região central essa taxa geralmente não ultrapassa os 85%. A mesma análise feita sobre a Figura 21 (c) mostra que este comportamento dos modos *skip* é semelhante para CUs  $32 \times 32$ : enquanto que na região superior e inferior do quadro a taxa de ocorrência de modos *skip* média é de aproximadamente 80%, na região central do quadro esta taxa é aproximadamente 70%.

Já a análise das taxas de ocorrência de *skip* para CUs  $64 \times 64$  na Figura 21 (d) mostra que existe um comportamento visivelmente mais heterogêneo nas CUs maiores. Para o caso das CUs  $64 \times 64$ , na região superior e extremo-inferior do quadro a taxa de ocorrência de modos *skip* é superior a 80%, enquanto que na região central esta taxa dificilmente ultrapassa 50%. Por fim, a Figura 21 (e) apresenta a taxa de ocorrência de modos *skip* para todos os tamanhos de CU combinados. A análise destes dados mostra que de fato, como as regiões superior e inferior são mais distorcidas – e consequentemente, redundantes – elas costumam ser codificadas com o modo *skip* com uma frequência maior do que o restante do quadro, e conforme aproxima-se da região central do quadro – onde a distorção é mínima – a taxa de ocorrência de modos *skip* é reduzida gradualmente.

As análises apresentadas deixam claro que durante a codificação de vídeos convencionais, as ocorrências do modo *skip* numa determinada região estão diretamente relacionadas ao conteúdo do vídeo nesta região. Por outro lado, quando vídeos omnidirecionais estão sendo codificados existe um comportamento espacial visível, em que as regiões superior e inferior do quadro são codificadas através do modo *skip* com uma frequência substancialmente maior do que as regiões centrais do quadro devido ao esticamento horizontal nas regiões superior e inferior.

### 5.2.2 Avaliação das Ocorrências de *skip* de Acordo com a Variância

Como mencionado anteriormente, o modo *skip* costuma ser utilizado em regiões muito homogêneas, e a homogeneidade criada nas regiões superior e inferior do quadro pela projeção ERP acarreta numa elevada utilização do modo *skip* nestas regiões. Dessa forma, foi feita uma segunda análise com o objetivo de correlacionar os dados de ocorrências de *skip* com a homogeneidade dos blocos que foram codificados com o modo *skip*. A variância das amostras do bloco foi utilizada para medir a homogeneidade, visto que a variância é uma boa métrica para avaliar a complexidade da textura deste bloco (ZATT et al., 2010).

Para realizar esta análise, foi extraído o tamanho e a posição de todos os blocos que foram codificados com o modo *skip* – assim como na avaliação anterior – e a variância de cada um destes blocos. Entretanto, enquanto que as sequências omnidirecionais *AerialCity*, *PoleVault* e *Broadway* são representadas com 8 bits para cada canal de cor, a sequência *SkateboardInLot* é representada com 10 bits para cada ca-

nal de cor. Dessa forma, a sequência *SkateboardInLot* foi descartada desta análise visto que por se tratar de uma sequência com faixa de representação muito maior, a variância pode assumir valores maiores e dominar as variâncias das sequências com 8 bits de representação. Em seguida, os dados de *skip* por variância de bloco foram divididos em cinco conjuntos de acordo com a posição vertical dos blocos dentro do quadro, seguindo a divisão em cinco bandas apresentada na Figura 19 para modelar o comportamento dos modos *skip* de acordo com as diferentes intensidades de distorção causada pela projeção ERP dentro do quadro.

Uma vez que os dados foram agrupados de acordo com as respectivas bandas, foi possível gerar a distribuição acumulada dos modos *skip* de acordo com suas variâncias, para cada banda e tamanho de bloco. No caso onde as variáveis são as ocorrências de *skip* e a variância do bloco, a distribuição acumulada pode ser interpretada da seguinte forma: para todo o universo dos *skips*  $S$  que ocorrem na banda  $B$  no tamanho de bloco  $T$ , a distribuição acumulada para uma variância  $V$  representa a percentagem dos *skips*  $S$  que ocorreram em blocos com uma variância menor ou igual a  $V$ .

Os resultados de distribuição de *skip* acumulada de acordo com a variância para CUs  $64 \times 64$  é apresentado na Figura 22 para vídeos convencionais, enquanto que os resultados para vídeos omnidirecionais são apresentados na Figura 23. Nestes gráficos, *Polar* representa a distribuição acumulada para as bandas polares superior e inferior, enquanto que *Meio-Polar* representa a distribuição acumulada para as bandas meio-superior e meio-inferior. Além disso, o eixo horizontal representa o valor de variância dos blocos enquanto que o eixo vertical representa o valor de distribuição acumulada. Como os vídeos foram avaliados considerando 8 bits para representar cada amostra, o valor máximo de variância é ligeiramente maior do que 16000, entretanto, os gráficos apresentam apenas a faixa entre as variâncias 0 e 1500 para facilitar a visualização. Além disso, as Figuras 22 e 23 apresentam os dados para CUs  $64 \times 64$  onde as variâncias das codificações com os quatro QPs avaliados (22, 27, 32 e 37) foram agrupadas.

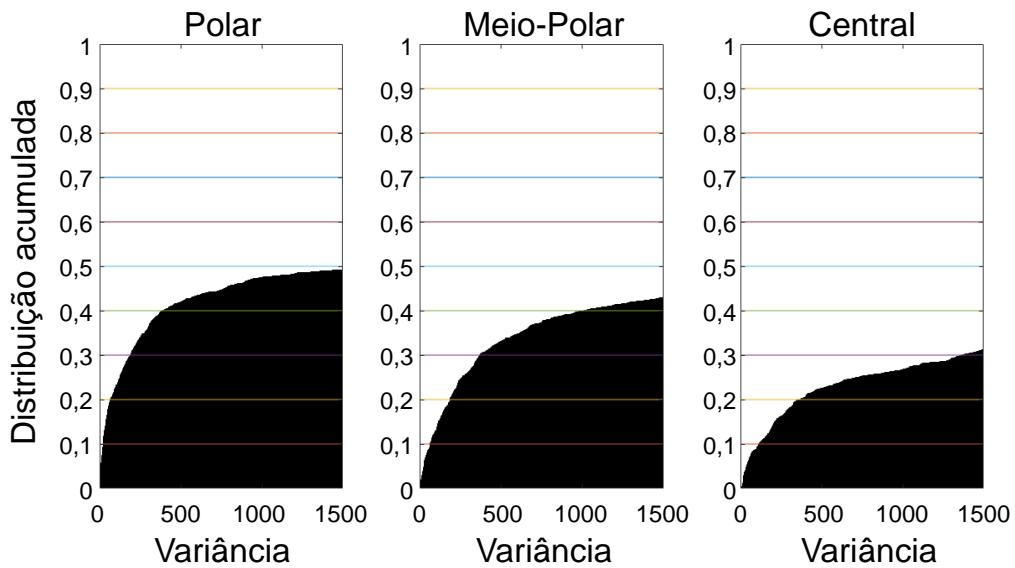


Figura 22 – Distribuição acumulada de *skips* de acordo com a variância de CUs  $64 \times 64$  em vídeos convencionais, com faixa de variância reduzida

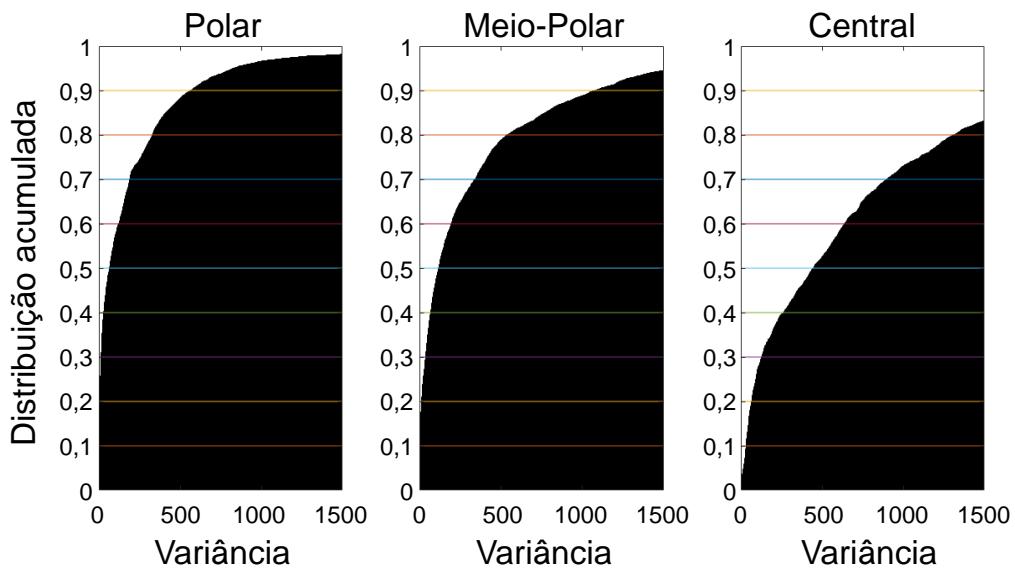


Figura 23 – Distribuição acumulada de *skips* de acordo com a variância de CUs  $64 \times 64$  em vídeos omnidirecionais, com faixa de variância reduzida

Quando os dados de distribuição acumulada para vídeos convencionais e omnidirecionais são avaliados, fica claro que esta distribuição tem um comportamento distinto para os diferentes tipos de vídeos e entre as bandas destes vídeos. Este comportamento pode ser observado ao fixar-se um valor de variância e observar a distribuição acumulada para a respectiva variância. Ao fixar-se um valor de variância igual a 500, por exemplo, obtém-se os seguintes resultados de distribuição acumulada.

No caso de vídeos convencionais, apresentados na Figura 22, nas bandas polares 41,94% dos *skips* ocorrem em blocos com variância menor ou igual a 500, enquanto

que nas bandas meio-polares 33,09% dos *skips* ocorrem nos blocos com variância menor ou igual a 500, e por fim, na banda central 22,54% dos *skips* ocorrem em blocos com variância menor ou igual a 500.

Para o caso de vídeos omnidirecionais, apresentados na Figura 23, nas bandas polares 88,19% dos *skips* ocorrem em blocos com variância menor ou igual a 500, enquanto que nas bandas meio-polares 78,79% dos *skips* ocorrem nos blocos com variância menor ou igual a 500, e por fim, na banda central apenas 52,45% dos *skips* ocorrem em blocos com variância menor ou igual a 500.

Estes resultados demonstram que durante a codificação de vídeos omnidirecionais, a grande maioria dos *skips* ocorrem em blocos com valores de variância razoavelmente pequenos – visto que para vídeos com 8 *bits*, a variância máxima ultrapassa 16000. Além disso, quando os resultados para as bandas de vídeos omnidirecionais são comparados, fica claro que existe um comportamento distinto entre as regiões com pouca distorção da banda central, e as regiões muito distorcidas das bandas polares e meio-polares. Enquanto que para a banda central pouco mais da metade dos *skips* ocorrem em blocos com variância menor ou igual a 500, nas bandas polares quase 90% dos *skips* ocorrem em blocos com variância até 500.

Além disso, a Figura 24 mostra a distribuição de *skips* acumulada para vídeos convencionais, novamente para CUs  $64 \times 64$  considerando todos os QPs, mas onde o eixo da variância compreende toda a faixa de valores possíveis, desde 0 até 16257. Além disso, os mesmos resultados para vídeos omnidirecionais são apresentados na Figura 25. Quando estes dados são analisados, uma nova característica distinta entre a codificação de vídeos convencionais e omnidirecionais pode ser observada.

Enquanto que para vídeos omnidirecionais a distribuição acumulada cresce de forma exponencial de acordo com a variância, até saturar em um, para vídeos convencionais a distribuição acumulada cresce de forma exponencial até aproximadamente 0,5 e 0,6, e a partir deste ponto a distribuição cresce de forma quase linear com a variância. Esse comportamento mostra que, além de apresentar uma distribuição mais acumulada nas baixas variâncias, a distribuição de *skips* para vídeos omnidirecionais também apresenta um formato diferente da distribuição de vídeos convencionais, apontando comportamentos distintos.

### 5.3 Avaliação dos Vetores de Movimento da Busca Raster

Como a projeção ERP estica as texturas horizontalmente com base na latitude das amostras e na função trigonométrica cosseno, é possível que trajetórias retilíneas na superfície esférica sejam mapeadas em trajetória curvas no vídeo projetado. Sendo assim, esta seção busca avaliar a **Hipótese 2** deste trabalho, isto é, a projeção ERP estica as texturas da superfície esférica e isso altera a magnitude e orientação dos

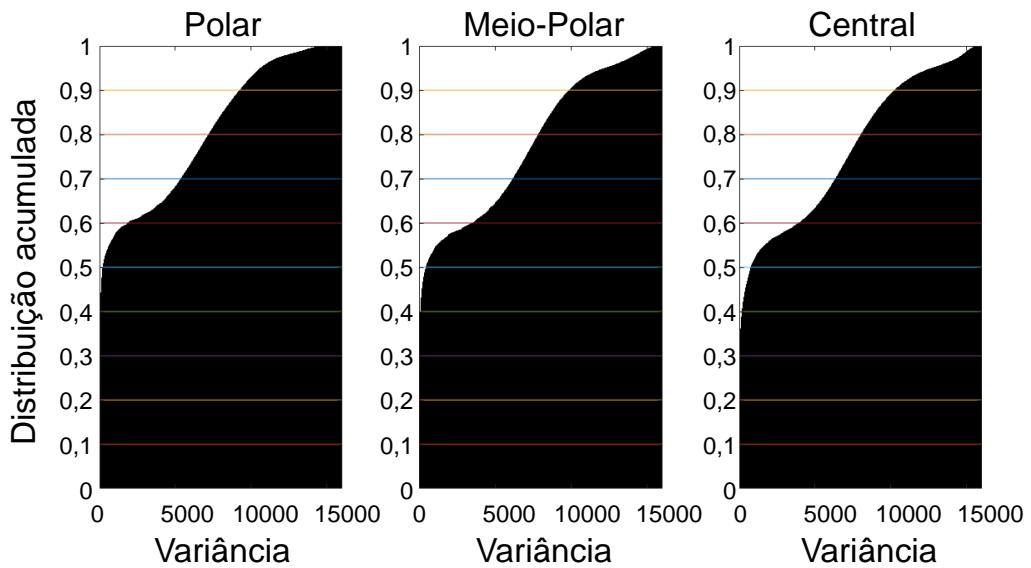


Figura 24 – Distribuição acumulada de *skips* de acordo com a variância de CUs  $64 \times 64$  em vídeos convencionais, com faixa de variância total

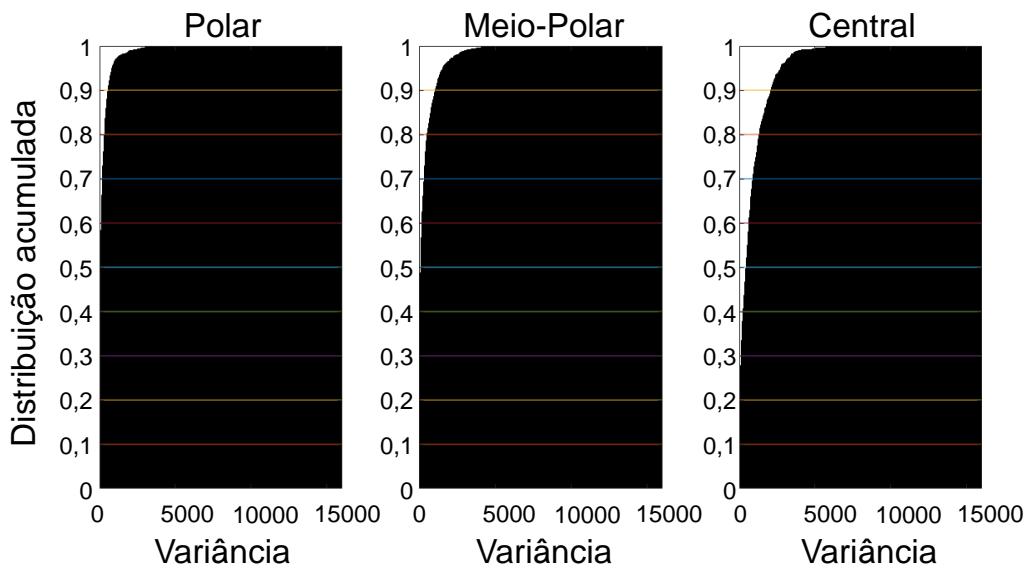


Figura 25 – Distribuição acumulada de *skips* de acordo com a variância de CUs  $64 \times 64$  em vídeos omnidirecionais, com faixa de variância total

vetores de movimento nas regiões distorcidas.

A estimativa de movimento no HM-16.16 é realizada através do algoritmo TZS, como descrito na Seção 3.3.3. O algoritmo TZS é composto por várias etapas e o trabalho (GONÇALVES et al., 2017) realizou um levantamento da complexidade de cada uma destas etapas e com qual frequência cada etapa encontrou um vetor de movimento ótimo. Esta avaliação indicou que as etapas de *Predição de Vetores de Movimento* e *Busca Inicial* representam aproximadamente 12% da complexidade total do algoritmo TZS e são responsáveis por encontrar 98% dos vetores de movimento ótimos, enquanto que a *Busca Raster* corresponde a 81% da complexidade total do

TZS e é responsável por encontrar apenas 0,4% dos vetores de movimento ótimos. A partir disso, fica claro que a etapa de *Busca Raster* é a mais suscetível a ser otimizada através de técnicas de redução de complexidade devido a alta complexidade desta etapa e a sua pequena participação nas decisões do codificador. Sendo assim, apenas os vetores de movimento resultantes da *Busca Raster* são avaliadas neste trabalho.

Para obter um entendimento sobre o comportamento da *Busca Raster* durante a codificação de vídeos omnidirecionais, três avaliações distintas foram realizadas. A primeira delas avaliou a distribuição angular dos vetores de movimento, e é descrita na Seção 5.3.1. A segunda avaliou a magnitude dos vetores de movimento, e é apresentada na Seção 5.3.2. Por fim, a terceira avaliação mediu o impacto da redução da área de busca, e é discutida na Seção 5.3.3.

### 5.3.1 Avaliação da Distribuição Angular dos Vetores da Busca Raster

Como a *Busca Raster* não é realizada para todos os blocos – como apresentado pelo fluxograma da Figura 16 – e quando executada, pode ter seu resultado atualizado por decisões de outras etapas da predição, o HM-16.16 (JCT-VC, 2017) foi modificado para permitir a extração dos vetores de movimento escolhidos durante a *Busca Raster* da codificação. Assim como na avaliação das ocorrências de *skip* na Seção 5.2, a avaliação dos vetores de movimento extraiu os vetores provenientes da *Busca Raster* para todos os tamanhos de bloco intermediários durante a codificação visto que isso representa as decisões ótimas locais.

De posse destes dados, foi realizada uma avaliação para medir a distribuição angular dos vetores de movimento resultantes da *Busca Raster* para determinar se estes vetores de movimento têm uma tendência a apresentar alguma direção específica. Como esta avaliação busca comparar as direções vertical/horizontal e não os sentidos cima/baixo e esquerda/direita, todos os vetores de movimento foram normalizados para o primeiro quadrante, isto é, para possuírem inclinação em relação ao eixo X entre 0° e 90°.

Em seguida, os vetores de movimento foram separados em três conjuntos de acordo com a abordagem em três bandas apresentada na Figura 19. Em seguida, os vetores de movimento de cada banda foram agrupados de acordo com suas inclinações  $\theta$  em relação ao eixo X, em classes que compreendem 15° cada, desde 0° até 90°. Vetores de movimento zero, isto é, onde tanto a componente X quanto Y são iguais a zero foram desconsiderados durante esta avaliação. A Tabela 3 representa a definição de cada classe de vetor de movimento, onde a coluna *Faixa angular* representa a faixa de inclinações compreendida por cada *Classe*.

A partir destas avaliações, as Figuras 26 e 27 apresentam a distribuição angular acumulada para vídeos convencionais e omnidirecionais, respectivamente. Nestas

Tabela 3 – Faixa angular de cada classe de vetor de movimento

Classe	Faixa angular
1	$0 \leq \theta_1 < 15$
2	$15 \leq \theta_2 < 30$
3	$30 \leq \theta_3 < 45$
4	$45 \leq \theta_4 < 60$
5	$60 \leq \theta_5 < 75$
6	$75 \leq \theta_6 \leq 90$

distribuições, todos os vetores de movimento obtiveram peso unitário, independente de sua magnitude, tanto para vídeos convencionais quanto omnidirecionais. Além disso, as barras em azul escuro representam os vetores com inclinação dentro de  $\theta_1$ , as barras laranjas representam os vetores com inclinação dentro de  $\theta_2$ , as barras cinzas representam os vetores com inclinação dentro de  $\theta_3$ , as barras amarelas representam os vetores com inclinação dentro de  $\theta_4$ , as barras em azul claro representam os vetores com inclinação dentro de  $\theta_5$ , e as barras verdes representam os vetores com inclinação dentro de  $\theta_6$ .

Quando os resultados de distribuição de inclinação para vídeos convencionais da Figura 26 são analisados, um comportamento bastante homogêneo fica evidente. Ao comparar a distribuição de inclinação para a banda superior e central, percebe-se que estas apresentam a mesma distribuição de vetores de movimento, isto é, a probabilidade de que um vetor de movimento esteja numa determinada faixa de inclinação é igual para ambas as regiões do vídeo. Quando estas bandas são comparadas com a banda inferior, uma pequena alteração é visível: na banda inferior a chance de um vetor de movimento ter inclinação entre  $0^\circ$  e  $15^\circ$  ou ter inclinação entre  $60^\circ$  e  $75^\circ$  é 1% menor do que nas bandas central e superior, enquanto que a probabilidade deste vetor de movimento ter inclinação entre  $75^\circ$  e  $90^\circ$  é 2% maior do que nas bandas central e superior. Estas variações na distribuição de inclinação dos vetores de movimento são tão pequenas que é possível considerá-las desprezíveis.

Estes dados também demonstram que durante a codificação de vídeos convencionais há uma tendência à escolha de vetores de movimento na direção horizontal, visto que a faixa de  $0^\circ$  a  $30^\circ$  corresponde a aproximadamente 50% dos vetores de movimento, para as três bandas. Estes dados apontam que além das três bandas apresentarem um comportamento similar, existe uma tendência para a escolha de vetores de movimento orientados na horizontal.

Por outro lado, a análise da distribuição de inclinações para vídeos omnidirecionais da Figura 27 mostra um comportamento significativamente diferente. Quando a faixa de inclinação entre  $75^\circ$  e  $90^\circ$  é avaliada, por exemplo, ela corresponde a 29% dos vetores de movimento resultantes da Busca Raster na banda superior, 16% dos

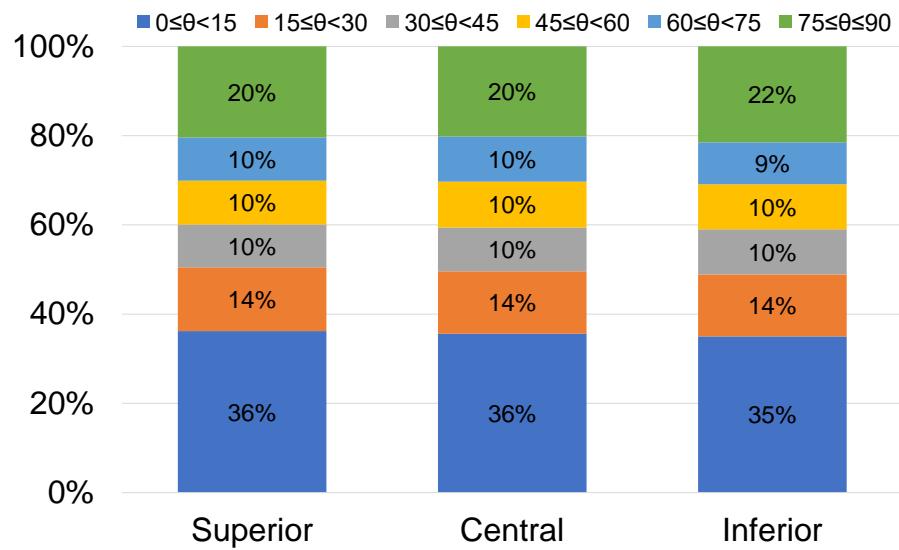


Figura 26 – Distribuição de inclinação dos vetores de movimento da Busca Raster para vídeos convencionais

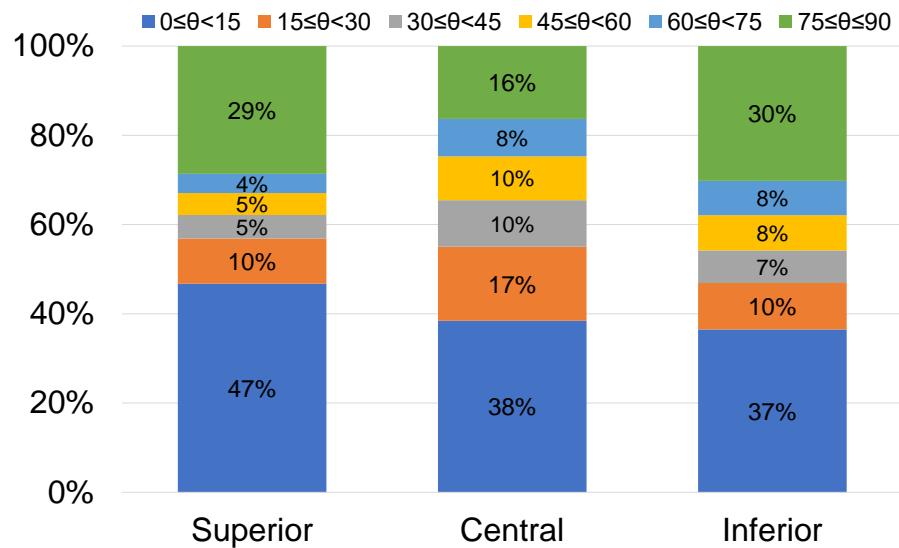


Figura 27 – Distribuição de inclinação dos vetores de movimento da Busca Raster para vídeos omnidirecionais

vetores de movimento da banda central e 30% dos vetores de movimento da banda inferior. Isso mostra que para esta faixa de inclinação, as bandas polares tem uma taxa de ocorrência muito semelhante, enquanto que a banda central tem um taxa de ocorrência significativamente menor. Entretanto, a mesma análise realizada sobre a faixa de inclinação entre 0° e 15° mostra um comportamento diferente: enquanto que esta faixa de inclinação corresponde a 47% dos vetores de movimento da banda superior, nas bandas central e inferior esta faixa de inclinação corresponde a 38% e 37% dos vetores de movimento, respectivamente. Isso mostra que a distribuição de inclinações nos vetores de movimento para as bandas polares não é semelhante em todos os aspectos, e em alguns casos, uma banda polar se comporta de forma mais semelhante

à banda central do que a sua banda polar oposta.

Desta forma, embora estes resultados mostrem que existem diferenças na direção dos vetores de movimento para vídeos convencionais e omnidirecionais, também percebe-se que as direções dos vetores de movimento de vídeos omnidirecionais não apresentam um comportamento tão regular como aquele de vídeos convencionais. Isso fica visível ao analisar-se as bandas superior e inferior de vídeos omnidirecionais, que mesmo apresentando a mesma intensidade de distorção causada pela projeção ERP, ainda possuem variações na distribuição de direções dos vetores de movimento.

### 5.3.2 Avaliação da Magnitude dos Vetores de Movimento

Para realizar a avaliação da magnitude dos vetores de movimento resultantes da Busca Raster, a mesma metodologia da avaliação angular foi realizada, isto é, os vetores de movimento resultante das decisões intermediárias do codificador foram extraídos e em seguida os vetores foram agrupados em três conjuntos de acordo com a abordagem em três bandas da Figura 19.

Em seguida, o valor médio e o desvio padrão da componente X e da componente Y dos vetores de movimento foi calculado para cada uma das três bandas com o objetivo de determinar se as distorções causadas pela projeção ERP interferem na magnitude e variabilidade dos vetores de movimento. Os resultados desta avaliação para vídeos convencionais e omnidirecionais são apresentados na Tabela 4.

Quando os dados referentes a vídeos convencionais são analisados, percebe-se que na banda central as componentes X e Y dos vetores de movimento tem valores médios de 15,0 e 9,5, respectivamente. Já na banda superior, a média das componentes X e Y é 16,0 e 9,3, respectivamente, enquanto que na banda inferior a média das componentes X e Y é 16,8 e 10,3, respectivamente. Esses dados mostram que a componente X dos vetores de movimento costuma ser maior do que a componente Y independente da região sendo codificada, além de mostrar que a média das componentes X e Y apresentam um comportamento recorrente nas diversas bandas: a componente X tem média 16,0, 15,0 e 16,8 nas bandas superior, central e inferior,

Tabela 4 – Média e desvio padrão das componentes X e Y dos vetores de movimento da Busca Raster

		Convencionais		Omnidirecionais	
		X	Y	X	Y
<b>Superior</b>	<b>Media</b>	16,0	9,3	9,8	4,5
	<b>Desvio</b>	24,4	14,9	15,2	8,9
<b>Central</b>	<b>Media</b>	15,0	9,5	15,1	8,5
	<b>Desvio</b>	24,0	16,2	19,8	13,8
<b>Inferior</b>	<b>Media</b>	16,8	10,3	12,5	10,9
	<b>Desvio</b>	26,4	16,4	21,3	19,3

respectivamente, enquanto que a componente Y tem média 9,3, 9,5 e 10,3 nas bandas superior, central e inferior, respectivamente. Dessa forma, pode-se dizer que a Busca Raster tem um comportamento similar para qualquer que seja a região do quadro sendo codificada.

Já ao analisar os dados referentes a codificação de vídeos omnidirecionais, percebe-se que na banda central a média das componentes X e Y dos vetores de movimento provenientes da Busca Raster é igual a 15,1 e 8,5, respectivamente. Na banda superior, por outro lado, a média das componentes X e Y é igual a 9,8 e 4,5, respectivamente, enquanto que na banda inferior a média das componentes X e Y é igual a 12,5 e 10,9, respectivamente. Esses resultados mostram que, assim como acontece com vídeos convencionais, a componente X dos vetores de movimento provenientes da Busca Raster costuma ter magnitude maior do que a componente Y.

Por outro lado, diferente do que acontece durante a codificação de vídeos convencionais, uma grande variabilidade é percebida quando uma mesma componente é analisada nas diferentes bandas. No caso da componente X, esta apresenta valor médio igual a 9,8, 15,1 e 12,5 nas bandas superior, central e inferior, respectivamente, mostrando que a componente X da banda central tem um valor médio mais de 50% maior do que a componente X na banda superior. O mesmo comportamento é percebido para a componente Y nas diferentes bandas, visto que na banda inferior de vídeos omnidirecionais a componente Y tem valor médio igual a 10,9, que é mais do que o dobro da média da componente Y na banda superior (que possui média igual a 4,5).

Estes resultados indicam um comportamento semelhante ao extraído da distribuição de inclinações nas Figuras 27 e 26, isto é, em vídeos convencionais os vetores de movimento provenientes da Busca Raster tem um comportamento semelhante em todas as bandas, enquanto que durante a codificação de vídeos omnidirecionais os vetores de movimento tem um comportamento diferenciado de acordo com a banda sendo codificada, e existem variações entre o comportamento da banda superior e inferior.

Entretanto, os valores de desvio padrão bastante elevados apontam que existe uma variabilidade muito grande nos vetores de movimento, tanto para vídeos convencionais quanto omnidirecionais. Com base nisso, é possível que os valores médios encontrados não refletem as tendências nas escolhas dos vetores de movimento, e não se sabe como a escolha de outros vetores pode impactar na eficiência de codificação.

### 5.3.3 Avaliação do Impacto da Redução da Área de Busca

Diferente das avaliações anteriores, onde a codificação foi realizada normalmente e as decisões do codificador foram avaliadas, esta análise busca medir o impacto causado no tempo e eficiência de codificação quando as escolhas de vetores de mo-

vimento são limitadas, isto é, existe uma interferência nas escolhas do codificador.

Para isso, o HM-16.16 (JCT-VC, 2017) foi modificado para que seja possível alterar as dimensões da área de busca da etapa de Busca Raster de acordo com a região do quadro sendo codificada.

A avaliação do impacto no tempo e eficiência de codificação foi realizada seguindo a abordagem em três bandas, nas proporções apresentadas na Figura 19. Além disso, foram realizados três conjuntos de codificações: no primeiro conjunto o codificador sem modificações foi utilizado; no segundo conjunto, o codificador foi alterado para limitar a altura da área de busca da Busca Raster para 30% da altura original nas bandas polares; e no terceiro conjunto o codificador foi alterado para limitar a largura da área de busca da Busca Raster para 30% da largura original nas bandas polares. As dimensões originais da área de busca no HM-16.16 (JCT-VC, 2017) são  $384 \times 384$ .

A Tabela 5 apresenta os resultados obtidos pela limitação da área de busca na Busca Raster para vídeos convencionais e omnidirecionais, onde a coluna *Vert 30%* representa o caso em que a área de busca foi reduzida verticalmente, e a coluna *Hori 30%* representa o caso em que a área de busca foi reduzida horizontalmente. Além disso, a coluna *RC* representam a redução de complexidade e a coluna *BD-BR* representa o BD-BR da codificação.

Quando os dados para vídeos convencionais são analisados, percebe-se que reduzir a área de busca verticalmente causou uma redução de complexidade média igual a 1,56%, enquanto que reduzir a área de busca horizontalmente causou uma redução de complexidade média igual a 1,83%. Isso mostra que mesmo ao reduzir-se a área de busca na mesma intensidade nos dois sentidos, reduções de complexidade distintas são observadas. Isso ocorre devido a interdependência entre as etapas do TZS: ao limitar a área de busca da Busca Raster, é possível que esta etapa retorne um

Tabela 5 – Redução de complexidade e eficiência de codificação na alteração da área de Busca Raster nas bandas polares

Tipo	Vídeo	Vert 30%		Hori 30%	
		RC	BD-BR	RC	BD-BR
Convencionais	<i>NebutaFestival</i>	0,69%	-0,05%	1,48%	-0,05%
	<i>PeopleOnStreet</i>	3,98%	-0,06%	3,86%	-0,02%
	<i>SteamLocomotiveTrain</i>	1,27%	-0,20%	1,45%	-0,15%
	<i>Traffic</i>	0,31%	0,01%	0,54%	-0,01%
	<b>Média Convencionais</b>	<b>1,56%</b>	<b>-0,07%</b>	<b>1,83%</b>	<b>-0,06%</b>
Omnidirecionais	<i>AerialCity</i>	0,22%	+0,00%	12,88%	-0,02%
	<i>Broadway</i>	3,16%	0,00%	3,40%	0,02%
	<i>PoleVault</i>	0,15%	0,01%	0,39%	-0,02%
	<i>SkateboardInLot</i>	5,00%	-0,02%	5,11%	-0,07%
	<b>Média omnidirecionais</b>	<b>2,13%</b>	<b>0,00%</b>	<b>5,44%</b>	<b>-0,02%</b>

vetor de movimento sub-ótimo para a etapa de Refinamento do TZS, e o Refinamento precise de mais iterações até atingir uma condição de parada. Além disso, o vetor de movimento escolhido para um bloco influencia na Predição de Vetores de Movimento dos próximos blocos, que pode causar diferentes alterações na complexidade do TZS. Já no aspecto eficiência de codificação, reduzir a área de busca verticalmente levou a um BD-BR igual a -0,07%, enquanto que a limitação da área de busca no sentido horizontal causou um BD-BR de -0,06%. Estes resultados mostram que é possível aumentar a eficiência de codificação mesmo ao limitar-se a área de busca, visto que isso leva a distribuições de vetores de movimento mais homogêneas e que podem ser codificados com menos *bits*. Além disso, o impacto na eficiência de codificação é similar independente se a área de busca é limitada no sentido vertical ou horizontal.

A análise dos dados referentes a vídeos omnidirecionais apresenta algumas semelhanças em relação aos vídeos convencionais. No aspecto eficiência de codificação, a limitação da área de busca no sentido vertical não alterou a eficiência de codificação (causou um BD-BR igual a 0%) enquanto que a limitação no sentido horizontal levou a um BD-BR de -0,02%. Isso mostra que assim como acontece com vídeos convencionais, é possível aumentar a eficiência de codificação ao limitar-se a área de busca, e além disso o impacto na eficiência de codificação é similar independente se a área de busca é limitada no sentido vertical ou horizontal. Entretanto, os resultados de redução de complexidade para vídeos omnidirecionais apresentam resultados distintos daqueles de vídeos convencionais. Quando a área de busca foi limitada no sentido vertical, uma redução de complexidade de 2,13% foi observada, enquanto que a limitação da área de busca na horizontal levou a uma redução de complexidade de 5,44%. A partir disso, percebe-se que a redução de complexidade foi maior para vídeos omnidirecionais em ambas as abordagens, e que a limitação da área de busca no sentido horizontal levou a uma redução de complexidade superior ao dobro da redução de complexidade atingida ao limitar-se a área de busca na direção vertical.

Estes resultados apontam que enquanto para vídeos convencionais, a limitação da área de busca no sentido vertical e horizontal levam a resultados semelhantes de redução de complexidade e eficiência de codificação, durante a codificação de vídeos omnidirecionais a limitação da área de busca no sentido horizontal leva a uma redução de complexidade maior. Isso pode ocorrer porque mesmo com uma área de busca horizontalmente menor, durante a codificação de vídeos omnidirecionais a Busca Raster é capaz de encontrar o vetor de movimento ótimo ou um vetor sub-ótimo que não interfere de forma significativa nas próximas etapas da codificação.

## 5.4 Avaliação do Impacto da Estimação de Movimento Fracionária

Quando a projeção ERP é utilizada, as texturas da superfície esférica são esticadas horizontalmente num fator proporcional a proximidades dos polos durante o mapeamento para a superfície retangular. Nesse processo, são geradas amostras em posições fracionárias na superfície esférica para preencher toda a largura do quadro retangular do vídeo num processo similar ao feito durante a FME.

Quando esta característica da projeção ERP é levada em consideração, percebe-se que a FME pode estar gerando amostras redundantes em algumas regiões do quadro. Tendo como exemplo o sistema de coordenadas polares apresentado na esquerda da Figura 28, onde a posição de uma amostra com dimensão unitária é determinada por  $\lambda$  e  $\phi$ , o mapeamento desta amostra unitária dá origem a um conjunto de amostras com dimensão vertical 1 e horizontal  $k$ , sendo que o esticamento horizontal  $k$  causado nas amostras de elevação  $\phi$  segue a Equação 2 (LUZ, 2017).

$$k = \frac{1}{\cos(\phi)} \quad (2)$$

Sendo assim, ao considerar-se as amostras que estão na elevação  $\phi$  igual a  $70^\circ$  na superfície da esfera o esticamento é aproximadamente 3. Dessa forma, cada amostra unitária na superfície da esfera, na elevação  $\phi$  igual a  $70^\circ$ , origina aproximadamente 3 amostras na superfície retangular, isto é, são geradas duas novas amostras em posições fracionárias entre cada dois *pixels* em posições inteiras. Neste caso, dois *pixels* em posições inteiras  $p_1$  e  $p_2$  na superfície esférica dão origem, além destes mesmos, aos *pixels* em posições fracionárias  $p_{1,33}$  e  $p_{1,66}$  antes de serem mapeados para a superfície retangular.

A partir disso, quando a FME é realizada na região equivalente à elevação de  $70^\circ$  no vídeo projetado, embora a FME tome como referência as amostras em posições inteiras dentro do vídeo projetado, algumas destas amostras representam posições fracionárias na superfície da esfera. Tomando como exemplo o *pixel* no vídeo projetado que corresponde ao *pixel* na posição horizontal  $p_{1,33}$  na superfície esférica,

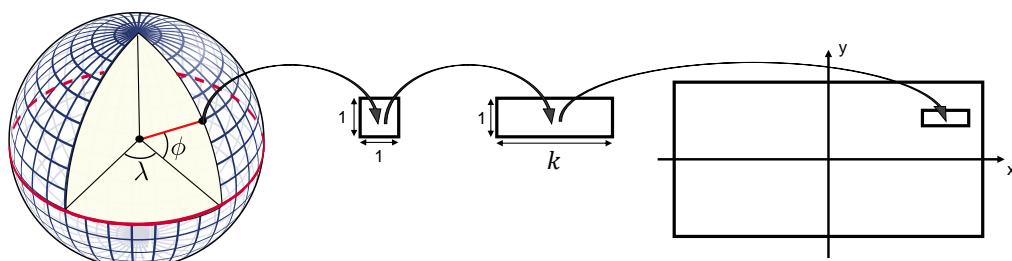


Figura 28 – Mapeamento de uma amostra unitária da superfície esférica para o plano. Adaptado de (LUZ, 2017)

quando são geradas amostras fracionárias com precisão de  $1/2$  pixel no vídeo projetado, estas amostras correspondem às amostras nas posições horizontais  $p_{1,17}$  (entre as amostras  $p_1$  e  $p_{1,33}$ ) e  $p_{1,50}$  (entre as amostras  $p_{1,33}$  e  $p_{1,66}$ ) na superfície esférica, isto é, amostras fracionárias com precisão de  $1/6$  pixel na superfície esférica.

A partir destas observações, esta seção busca avaliar qual é o ganho de qualidade acarretado pela realização da FME na codificação de vídeos omnidirecionais na projeção ERP, isto é, avaliar a **Hipótese 3** deste trabalho. Assim como foi feito durante a avaliação do impacto da redução da área de busca na Subseção 5.3.3, para avaliar o impacto da FME foi necessário interferir na codificação para obter resultados. Isso se deve porque a FME é realizada para todos os blocos, e embora ela possa ou não originar um vetor de movimento mais adequado para o bloco em questão, isso não mede o ganho de eficiência de codificação quando comparado a uma codificação sem FME.

Dessa forma, o HM-16.16 (JCT-VC, 2017) foi modificado para permitir a limitação na precisão da FME tanto no sentido vertical quanto horizontal. A partir disso, é possível realizar a FME com precisões diferentes em direções diferentes, como por exemplo, avaliar posições verticalmente fracionárias com precisões de  $1/2$  e  $1/4$  pixel e horizontalmente fracionárias com precisão máxima de  $1/2$  pixel, ou até mesmo não realizar a FME em alguma direção.

No HM original a FME é realizada em duas etapas para cada precisão: primeiro as amostras nas posições fracionárias são geradas, e em seguida, estas amostras são avaliadas para determinar qual atinge os melhores resultados. Sendo assim, visto que em alguns casos apenas um subconjunto das amostras em posições fracionárias são avaliadas, o HM também foi modificado para gerar apenas as amostras fracionárias nas posições que serão avaliadas posteriormente, evitando assim computações desnecessárias e aumentando a redução de complexidade.

A partir desta FME com precisão adaptativa, o conjunto de vídeos convencionais da Tabela 2 e de vídeos omnidirecionais da Tabela 1 foram codificadas utilizando a abordagem em três bandas da Figura 19. Além disso, as codificações se deram com três conjuntos de parâmetros. No primeiro conjunto, as sequências foram codificadas de acordo com o HM original, isto é, a FME foi realizada de forma completa em todo o quadro dos vídeos. No segundo conjunto, a banda central foi codificada com a FME completa, enquanto que nas bandas polares a FME foi aplicada apenas no sentido horizontal, utilizando as precisões de  $1/2$  e  $1/4$  pixel. Por fim, no terceiro conjunto a banda central foi codificada com a FME completa novamente, enquanto que nas bandas polares a FME foi aplicada apenas no sentido vertical, utilizando as precisões de  $1/2$  e  $1/4$  pixel.

A partir destes três conjuntos de simulações, foi avaliado o quanto a não-realização da FME interfere no tempo de codificação e na eficiência de codificação de vídeos

Tabela 6 – Redução de complexidade e eficiência de codificação na limitação de precisão da FME

Tipo	Vídeo	Polar Horizontal		Polar Vertical	
		RC	BD-BR	RC	BD-BR
Convencionais	<i>NebutaFestival</i>	5,49%	0,67%	6,70%	0,54%
	<i>PeopleOnStreet</i>	6,43%	1,83%	7,74%	1,57%
	<i>SteamLocomotiveTrain</i>	7,44%	2,77%	9,31%	2,77%
	<i>Traffic</i>	7,47%	2,45%	10,60%	2,69%
	<b>Média convencionais</b>	<b>6,71%</b>	<b>1,93%</b>	<b>8,89%</b>	<b>1,89%</b>
Omnidirecionais	<i>AerialCity</i>	8,56%	1,31%	10,68%	0,75%
	<i>PoleVault</i>	6,19%	1,65%	7,92%	1,02%
	<i>Broadway</i>	7,58%	0,40%	9,52%	0,26%
	<i>SkateboardInLot</i>	6,19%	0,24%	7,59%	0,13%
	<b>Média omnidirecionais</b>	<b>7,13%</b>	<b>0,90%</b>	<b>8,93%</b>	<b>0,54%</b>

omnidirecionais e convencionais.

Os resultados de redução de complexidade e eficiência de codificação destas avaliações são apresentados na Tabela 6, onde as colunas *Polar Horizontal* e *Polar Vertical* correspondem aos resultados para o segundo e terceiro conjunto de codificações, respectivamente. Dessa forma, em *Polar Vertical* a FME não foi realizada no sentido horizontal e em *Polar Horizontal* a FME não foi realizada no sentido vertical. Além disso, a coluna *RC* corresponde aos resultados de redução de complexidade, enquanto que *BD-BR* corresponde ao BD-BR atingido pela codificação.

De acordo com os resultados de redução de complexidade da Tabela 6, durante a codificação de vídeos convencionais realizar a FME apenas na direção horizontal nas regiões polares (coluna *Polar Horizontal*) acarreta uma redução de complexidade igual a 6,71%, enquanto que realizar a FME apenas na direção vertical nas regiões polares (coluna *Polar Vertical*) causa uma redução de complexidade média igual a 8,59%. Como a FME realiza o mesmo conjunto de cálculos para vídeos de mesmas resoluções, a variabilidade na redução de complexidade observada numa mesma coluna está associada ao comportamento de outros algoritmos que apresentam característica interativa ou recursiva e ao quanto a FME contribui para a complexidade total da codificação: nos casos onde a FME é mais significativa, reduzir sua complexidade leva a reduções de complexidade gerais maiores. Além disso, uma avaliação dos resultados para vídeos omnidirecionais mostra que realizar a FME apenas na direção horizontal nas regiões polares (coluna *Horizontal Polar*) nestes vídeos causa uma redução de complexidade média igual a 7,13%, enquanto que realizar a FME apenas na direção vertical nas regiões polares (coluna *Vertical Polar*) leva a uma redução de complexidade média igual a 8,93%. A variabilidade nos resultados de uma mesma coluna se dão pelo mesmo motivo descrito no vídeos convencionais.

A partir destas análises percebe-se que realizar a FME apenas na direção vertical

ou horizontal nas regiões polares do quadro, tanto durante a codificação de vídeos convencionais quanto omnidirecionais, leva a reduções de complexidade semelhantes. Sendo assim, é possível dizer que a FME apresenta uma complexidade semelhante para ambos os conjuntos de vídeos, e técnicas de redução de complexidade da FME tendem a obter resultados semelhantes.

Já quando os dados de eficiência de codificação da Tabela 6 são analisados, fica visível que a degradação causada em vídeos convencionais é semelhante independente de a FME ser realizada apenas no sentido vertical ou apenas no sentido horizontal, enquanto que para vídeos omnidirecionais a degradação é distinta.

De forma mais específica para vídeos convencionais, quando a FME é realizada apenas no sentido horizontal no conjunto *Polar Horizontal* observa-se uma redução na eficiência de codificação de 1,93% BD-BR, enquanto que quando a FME é realizada apenas no sentido vertical no conjunto *Polar Vertical* observa-se uma redução na eficiência de codificação de 1,89% BD-BR. Uma diferença de BD-BR desta magnitude indica que durante a codificação de vídeos convencionais, não realizar a FME no sentido vertical ou horizontal acarreta na mesma redução de eficiência de codificação.

Já no caso de vídeos omnidirecionais, quando a FME é realizada apenas no sentido horizontal no conjunto *Polar Horizontal* observa-se uma redução de eficiência de codificação de 0,90%, enquanto que quando a FME é realizada apenas no sentido vertical no conjunto *Polar Vertical* observa-se uma redução na eficiência de codificação de 0,54%. Diferente de vídeos convencionais, não realizar a FME no sentido horizontal acarreta uma redução na eficiência de codificação substancialmente menor do que não realizá-la no sentido vertical. Isso aponta que avaliar amostras em posições horizontalmente fracionárias nas regiões polares de vídeos omnidirecionais traz poucos ganhos de eficiência de codificação visto que muitas das amostras no vídeo projetado já são resultantes de posições fracionárias na superfície da esfera. Por outro lado, a projeção ERP não gera amostras em posições verticalmente fracionárias, dessa forma quando a FME é realizada no sentido vertical esta gera amostras genuinamente novas e fracionárias, aumentando assim a eficiência de codificação.

## 5.5 Avaliação dos Tamanhos de Bloco na Predição Intraquadro

Como descrito anteriormente, a projeção ERP causa distorções na textura do vídeo de acordo com a posição vertical dentro do quadro, de modo que nas regiões polares as texturas sofrem um esticamento horizontal severo enquanto que na região central do vídeo o esticamento é suave ou até mesmo nulo. Estas distorções fazem com que em algumas regiões do quadro as texturas tomem orientações mais horizontais e os blocos fiquem mais homogêneos.

No trabalho (STORCH et al., 2019) os autores realizaram uma avaliação do com-

portamento da predição intraquadro durante a codificação de vídeos omnidirecionais, e foi constatado que durante a codificação de blocos nas regiões polares, os modos de predição orientados na horizontal são escolhidos com uma frequência significativamente maior do que os demais modos de predição direcionais, enquanto que durante a codificação da região central do vídeo todos os modos de predição direcionais apresentam taxas de ocorrência semelhantes entre si. Dessa forma, foi comprovado que as distorções da projeção ERP afetam o comportamento da predição intraquadro. Entretanto, este trabalho não avalia como que a projeção ERP influencia nas decisões dos tamanhos de bloco. Dessa forma, esta seção busca investigar a **Hipótese 4** deste trabalho, isto é, o esticamento horizontal causado pela projeção ERP faz com que as regiões polares do vídeo sejam mais propícias a serem codificadas com blocos maiores.

Sendo assim, o conjunto de vídeos convencionais da Tabela 2 e omnidirecionais da Tabela 1 foram codificados com o HM-16.16 (JCT-VC, 2017) junto da 360Lib (YE; ALSHINA; BOYCE, 2017), e o tamanho de cada PU utilizada na codificação final foi extraída junto das suas respectivas posições. Vale lembrar que diferente das avaliações anteriores, esta avaliação lida com as decisões da predição intraquadro, e desta forma, as sequências convencionais utilizadas possuem resolução  $1920 \times 1080$ .

A partir dos dados extraídos, as informações foram agrupadas de acordo com a linha de CTUs dentro do quadro em que as respectivas PUs estão localizadas, isto é, todas as PUs na primeira linha de CTUs (amostras nas posições verticais desde 0 até 63) formaram um grupo, todas as PUs na segunda linha de CTUs (amostras nas posições verticais desde 64 até 123) formaram outro grupo, e assim por diante. A partir destes grupos, foi possível gerar uma distribuição de tamanhos de PU de acordo com a posição vertical dentro do quadro. Entretanto, como PUs de dimensões  $64 \times 64$  compreendem uma área equivalente a 256 PUs de dimensões  $4 \times 4$ , as ocorrências de cada tamanho de PU foram normalizadas para corresponder às respectivas áreas. Dessa forma, cada PU de dimensões  $4 \times 4$  obteve peso 1 nesta distribuição, cada PU de dimensões  $8 \times 8$  obteve peso 4, cada PU de dimensões  $16 \times 16$  obteve peso 16, cada PU de dimensões  $32 \times 32$  obteve peso 64, e cada PU de dimensões  $64 \times 64$  obteve peso 256.

Os resultados para a avaliação da distribuição de tamanhos de PU por linha de CTU para vídeos convencionais são apresentados na Figura 29, enquanto que os resultados para vídeos omnidirecionais são apresentados na Figura 30. Nestes gráficos, as barras na cor azul claro representam as PUs  $4 \times 4$ , as barras verdes representam PUs  $8 \times 8$ , as barras na cor azul escuro representam PUs  $16 \times 16$ , as barras vermelhas representam as PUs  $32 \times 32$  e as barras amarelas representam as PUs  $64 \times 64$ . Para cada tipo de vídeo são apresentados dois gráficos, um deles onde PUs  $4 \times 4$  estão no primeiro plano (Visão frontal), e outro onde PUs  $64 \times 64$  estão no primeiro plano

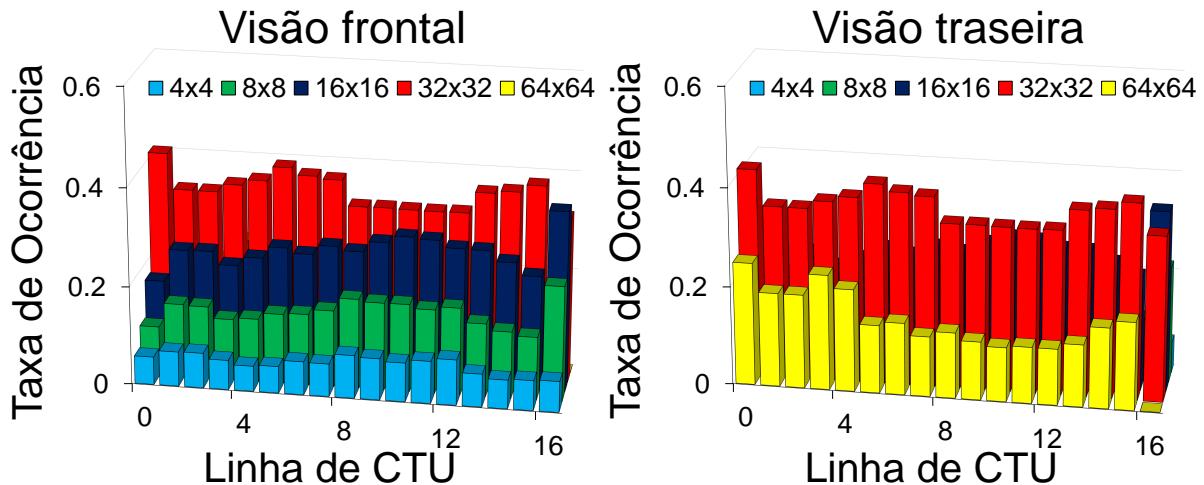


Figura 29 – Distribuição dos tamanhos de PU por linha de CTUs para vídeos convencionais. Adaptado de (STORCH et al., 2019)

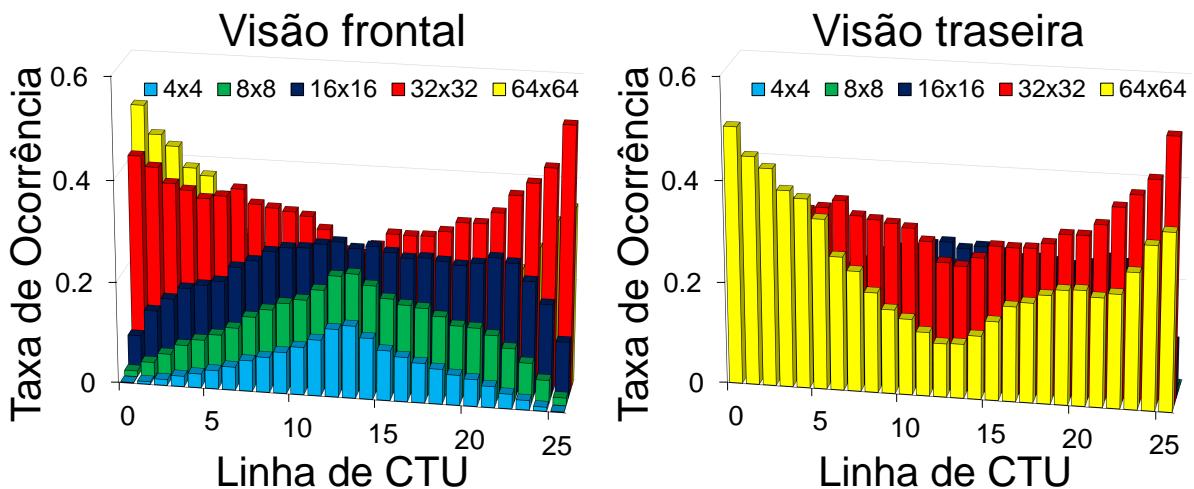


Figura 30 – Distribuição dos tamanhos de PU por linha de CTUs para vídeos omnidirecionais. Adaptado de (STORCH et al., 2019)

(Visão traseira). Além disso, o eixo horizontal indica as linhas de CTU, enquanto que o eixo vertical representa a taxa de ocorrência normalizada de cada tamanho de PU na respectiva linha de CTU. Como os tamanhos de PU são normalizados, a soma das ocorrências de todos os tamanhos de PU em qualquer linha de CTU sempre resulta em um.

Quando os dados de distribuição de tamanhos de PU por linha de CTU para vídeos convencionais da Figura 29 são analisados, fica claro que existe uma distribuição bastante homogênea para qualquer que seja a linha de CTU sendo codificada, isto é, independente da região do quadro sendo codificada, a probabilidade de um determinado tamanho de PU ser escolhido é sempre próxima de um mesmo determinado

valor. Dessa forma, pode-se concluir que PUs  $4 \times 4$  ocorrem com pouca frequência, e ao aproximar-se das PUs maiores esta frequência aumenta, com exceção das PUs  $64 \times 64$  que ocorrem com frequência similar as PUs  $16 \times 16$ . Além disso, nesta distribuição fica visível que PUs  $64 \times 64$  nunca são escolhidas na última linha de CTUs (linha 16). Isso ocorre porque os vídeos convencionais analisados possuem resolução vertical de 1080 amostras, o que resulta em 16,875 CTUs completas ( $1080 / 64$ ). Na prática o codificador trata isso como 17 CTUs e preenche a parte faltante com amostras pretas, e devido a esta borda bem definida entre a parte real e a parte preenchida da última CTU, PUs de dimensões  $64 \times 64$  nunca são escolhidas.

Já quando a distribuição dos tamanhos de PU por linha de CTU para vídeos omnidirecionais da Figura 30 é analisada, fica claro que existe um comportamento distinto na distribuição dos tamanhos de PU de acordo com a região do quadro sendo codificada.

Nas extremidades do quadro, isto é, próximo das linhas de CTU 0 e 25, as PUs de dimensões maiores são escolhidas na maioria das vezes, e conforme aproxima-se da região central do quadro, entre as linhas de CTU 10 e 15, as PUs maiores ocorrem com uma frequência cada vez menor, enquanto que a taxa de ocorrência de PUs menores aumenta.

Essa distribuição dos tamanhos de PU está diretamente relacionada ao esticamento horizontal causado pela projeção ERP: quanto mais próximo das regiões polares, mais intenso é o esticamento, logo, mais homogêneas são as texturas. Dessa forma, nas regiões polares PUs grandes são capazes de representar as regiões do vídeo com boa qualidade enviando poucas informações de resíduo, enquanto que nas regiões mais ao centro do quadro, onde há menor homogeneidade, é necessário empregar PUs menores para isolar as diferentes orientações de textura em diferentes blocos e atingir uma boa eficiência de codificação.

Com base nesse comportamento, fica visível que é possível desenvolver técnicas de redução de complexidade baseadas em descartar a avaliação de PU menores durante a codificação de algumas regiões do quadro, levando a uma redução no tempo de processamento destes vídeos sem causar um impacto significativo na qualidade visual dos mesmos.

## 5.6 Considerações Finais

Este capítulo apresentou uma avaliação do comportamento da codificação de vídeos omnidirecionais em relação a diversos aspectos, como a ocorrência do modo de codificação *skip*, como os vetores de movimento da Busca Raster se comportam nas diferentes regiões do quadro, o impacto da FME na qualidade de diferentes regiões do quadro e como a predição intraquadro escolhe os tamanhos de bloco a serem

utilizados.

Durante a avaliação do comportamento do modo de predição *skip*, os resultados apontam que vídeos omnidirecionais são codificados através do modo *skip* com uma frequência substancialmente maior do que vídeos convencionais. Além disso, as regiões superior e inferior dos quadros de vídeos omnidirecionais ainda empregam o modo *skip* com uma frequência maior do que a região central dos mesmos, visto que o esticamento causado pela projeção ERP é mais intensa nestas regiões.

No que diz respeito à etapa de Busca Raster do TZS, as avaliações conduzidas apontam que o esticamento horizontal causado pela projeção ERP faz com que os vetores de movimento tenham uma distribuição distinta nas regiões onde há maior e menor distorção, tanto em relação às direções quanto às magnitudes destes.

As avaliações do impacto da FME em vídeos omnidirecionais também apontam uma relação entre o esticamento causado pela projeção ERP e os ganhos de eficiência de codificação obtidos através da FME. Como muitas amostras nas regiões polares de vídeos omnidirecionais na projeção ERP são oriundas de posições fracionárias na superfície esférica, quando a FME é aplicada nestas amostras são geradas amostras em posições fracionárias redundantes que levam a pequenos ganhos de eficiência de codificação, indicando que a FME tem um impacto distinto na eficiência de codificação de acordo com a região do vídeo onde é aplicada.

Em relação aos tamanhos de bloco escolhidos durante a predição intraquadro, as avaliações apontaram que nas regiões superior e inferior do quadro, onde o esticamento causado pela projeção ERP é mais intenso, blocos de dimensões maiores tendem a ser escolhidas com muita frequência. Além disso, conforme aproxima-se da região central do quadro, que apresenta esticamento suave ou nulo, os tamanhos de bloco maiores tendem a ocorrer com menor frequência enquanto que as ocorrências de blocos menores se tornam mais significativas.

Estas descobertas a respeito da codificação de vídeos omnidirecionais serão exploradas no Capítulo 6 para dar origem a técnicas de redução de complexidade, que serão avaliadas no Capítulo 7.

## 6 TÉCNICAS DE REDUÇÃO DE COMPLEXIDADE PARA A CODIFICAÇÃO DE VÍDEOS OMNIDIRECIONAIS

As avaliações apresentadas no Capítulo 5 demonstraram que a codificação de vídeos omnidirecionais na projeção ERP apresenta uma série de características específicas quando comparada a codificação de vídeos convencionais, como uma tendência a utilizar blocos maiores e o modo *skip* nas regiões superior e inferior do quadro, além de que a FME apresenta ganhos de eficiência de codificação menores em algumas regiões do quadro e os vetores de movimento apresentam maior variabilidade nas diferentes regiões do quadro. A partir destas análises um conjunto de técnicas de redução de complexidade foi desenvolvido visando explorar essas características da codificação de vídeos omnidirecionais na projeção ERP para acelerar a codificação destes sem causar reduções significativas na eficiência de codificação.

A Figura 31 apresenta uma visão em alto nível de todas as técnicas desenvolvidas neste trabalho. Durante a codificação de um vídeo omnidirecional, o codificador se comunica com uma série de técnicas de redução de complexidade para determinar

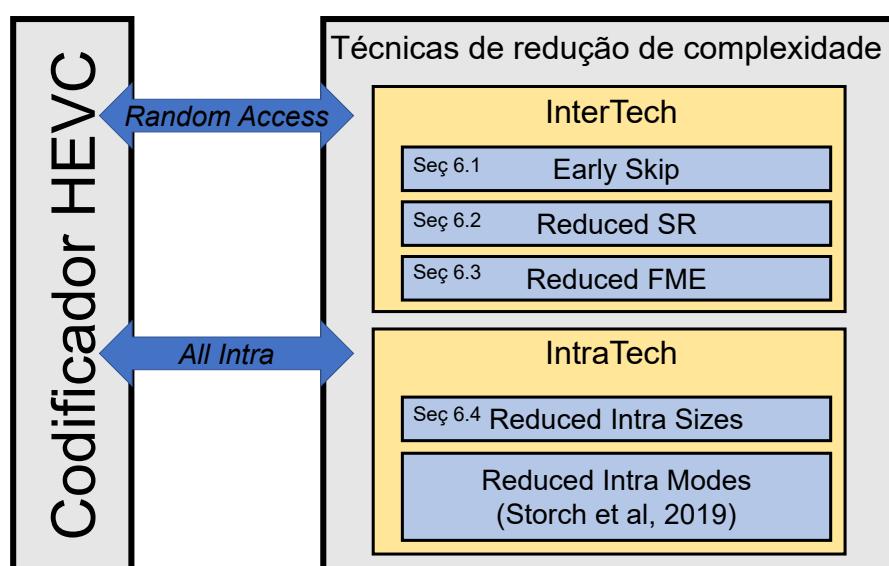


Figura 31 – Visão em alto nível das técnicas desenvolvidas junto ao codificador HEVC

se algumas etapas da codificação devem ser realizadas normalmente ou se devem ser simplificadas. Caso o codificador esteja utilizando a configuração de codificação *Random Access*, então a técnica *InterTech* é utilizada para acelerar a codificação, enquanto que durante a utilização da configuração *All Intra* a técnica *IntraTech* é utilizada.

A técnica *InterTech* é composta por três técnicas menores: *Early Skip*, *Reduced SR* e *Reduced FME*. A técnica *Early Skip* faz uma decisão prematura pelo modo de codificação *skip* baseado na região do quadro e variância do bloco, e é definida em detalhes na Seção 6.1. A técnica *Reduced SR* altera as dimensões da área de busca da Busca Raster de acordo com a região do quadro sendo codificada, e seus detalhes são descritos na Seção 6.2. A técnica *Reduced FME* reduz a precisão da FME no sentido horizontal de acordo com a região do quadro sendo codificada e é descrita na Seção 6.3.

Já a técnica *IntraTech* é composta por duas técnicas menores: *Reduced Intra Sizes* e *Reduced Intra Modes*. A técnica *Reduced Intra Sizes* descarta a avaliação de alguns tamanhos de PU de acordo com a região do quadro e o custo RD do bloco sendo codificado, e seu funcionamento é apresentado na Seção 6.4. A técnica *Reduced Intra Modes* reduz o número de modos de predição intraquadro avaliados de acordo com o tamanho de bloco e região do quadro sendo codificado. A técnica *Reduced Intra Modes* é produto de um trabalho anterior, e dessa forma, seu funcionamento não é detalhado neste capítulo mas sim no trabalho (STORCH et al., 2019). Por fim, a Seção 6.5 descreve o funcionamento de todas as técnicas atuando juntas.

## 6.1 *InterTech: Early Skip*

As avaliações conduzidas durante a Seção 5.2 apontaram que o modo de codificação *skip* é utilizado com maior frequência durante a codificação de vídeos omnidirecionais quando comparada a codificação de vídeos convencionais. Além disso, estas avaliações também mostraram que os *skips* ocorrem com maior frequência nas regiões polares de vídeos omnidirecionais e estas ocorrências de *skip* estão diretamente relacionadas a maior homogeneidade dos blocos situados nas regiões polares. Desta forma, esta seção propõe a técnica *Early Skip*, que busca acelerar a predição interquadros realizando uma decisão prematura pelo modo de codificação *skip* com base na homogeneidade do bloco sendo codificado. Para medir a homogeneidade dos blocos, a técnica *Early Skip* calcula a variância dos blocos.

Visto que a técnica *Early Skip* determina se um bloco deve ser codificado com o modo *skip* com base na variância do próprio bloco, surge a necessidade de um valor de corte para o qual a decisão pelo modo *skip* será feita de forma prematura. Estes valores são advindos de análises semelhantes àquelas conduzidas na Seção 5.2.

Como demonstrado anteriormente, os modos *skip* ocorrem em maior frequência nas regiões polares de vídeos omnidirecionais, e a distribuição de *skips* acumulada de acordo com a variância dos blocos também é mais concentrada nas baixas variâncias durante a codificação das regiões polares. Desta forma, os valores de variância de corte são gerados a partir de distribuições acumuladas de *skip* utilizando uma abordagem em bandas, seguindo as mesmas proporções apresentadas na Figura 19, isto é, uma abordagem com três bandas e proporções 25% - 50% - 25%, e uma abordagem com cinco bandas e proporções 15% - 20% - 30% - 20% - 15%.

Como indicado nas CTCs360 (BOYCE et al., 2017), vídeos omnidirecionais na projeção ERP com resolução 4k devem ser codificados na resolução  $3328 \times 1664$ , enquanto que vídeos com resolução 6k e 8k devem ser codificados na resolução  $4096 \times 2048$ . Uma CTU contém a mesma quantidade de dados em vídeos  $3328 \times 1664$  e  $4096 \times 2048$ , visto que as dimensões da CTU são constantes e iguais a  $64 \times 64$ . Entretanto, a quantidade de informação contida numa CTU pode variar de acordo com a resolução do vídeo: uma CTU representa 0,074% de um quadro  $3328 \times 1664$ , e 0,049% de um quadro  $4096 \times 2048$ . Desta forma, é possível que blocos de mesmo tamanho tenham mais bordas bem definidas em vídeos de resoluções menores, o que em geral leva à variâncias mais elevadas.

Sendo assim, os valores de variância para os quais o modo *skip* será escolhido de forma prematura são extraídos a partir de dois conjuntos de avaliações: um conjunto para vídeos codificados na resolução  $3328 \times 1664$  e outro para vídeos codificados na resolução  $4096 \times 2048$ . Para realizar o primeiro conjunto de avaliações, a distribuição acumulada de *skips* de acordo com a variância dos blocos é calculada para as sequências de teste *AerialCity*, *PoleVault* e *Broadway* (a sequência *Broadway* foi redimensionada para 4k e também codificada na resolução  $3328 \times 1664$ ). Já para o segundo conjunto de avaliações, a sequência *SkateboardInLot* é codificada na resolução  $4096 \times 2048$ .

Além da resolução, a variância de um determinado bloco também pode ser influenciada pelo tamanho do bloco em questão: é comum que uma CU  $8 \times 8$  compreenda uma região bastante homogênea e tenham baixa variância, enquanto que como CUs  $64 \times 64$  abrangem uma área maior elas normalmente contém mais detalhes, e dessa forma, apresentam maior variância. Desta forma, as distribuições de *skip* de acordo com a variância são realizadas para todos os tamanhos de CU individualmente:  $8 \times 8$ ,  $16 \times 16$ ,  $32 \times 32$  e  $64 \times 64$ .

É importante destacar que as sequências de teste fornecidas pelas CTCs360 (BOYCE et al., 2017) são compostas por 10 sequências com 8 *bits* de representação por cor e apenas 2 sequências com 10 *bits* de representação por cor, e a sequência *SkateboardInLot* é uma das sequências com 10 *bits* por cor. Como blocos representados com 10 *bits* por cor podem ter uma variância naturalmente maior do que blocos

com 8 *bits* por cor devido a maior faixa de representação, e uma pequena parcela das sequências possuem 10 *bits* por cor, as amostras nos blocos da sequência *SkateboardInLot* foram deslocadas 2 *bits* para a direita para calcular a variância com apenas 8 *bits*. Mesmo calculando a variância com 8 *bits*, as codificações são realizadas com os 10 *bits* originais.

Estas análises dão origem a uma série de distribuições acumuladas do modo *skip* de acordo com a variância dos blocos. Há um conjunto de distribuições acumuladas para a abordagem com três bandas e outro conjunto para a abordagem com cinco bandas. Para cada um destes conjuntos, ainda existem dois subconjuntos para vídeos codificados nas resoluções  $3328 \times 1664$  e  $4096 \times 2048$ . Dentro destes subconjuntos ainda existem quatro conjuntos menores para as distribuições de CUs  $8 \times 8$ ,  $16 \times 16$ ,  $32 \times 32$  e  $64 \times 64$ . Por fim, para cada um destes conjuntos menores, existem quatro distribuições para o caso da codificação estar utilizando QP igual a 22, 27, 32 ou 37, e outra distribuição para o caso da codificação utilizar outro valor de QP.

Estes conjuntos de distribuição se devem porque, como mencionado anteriormente, de acordo com a posição das bandas elas podem ser mais ou menos suscetíveis a serem codificadas com o modo *skip*. Além disso, a resolução das sequências e o tamanho dos blocos podem interferir na variância dos blocos. Por fim, as distribuições para QPs individuais são importantes porque quanto maior o QP empregado, mais o codificador prioriza a redução no *bitrate*, e a utilização do modo *skip* se torna mais frequente. Sendo assim, para QPs maiores o limiar de variância tende a ser mais baixo. A distribuição acumulada para o caso do codificador utilizar outro valor de QP é feita a partir dos dados de codificações com QPs 22, 27, 32 e 37, todos reunidos.

A partir de cada uma das distribuições acumuladas são gerados 18 pontos de operação, que caracterizam diferentes valores de distribuição de *skip* acumulada de acordo com a variância do bloco. Os pontos de operação são denominados *cutoffs* e variam desde 0,05 até 0,90, com passos de 0,05. Sendo assim, um mesmo valor de *cutoff* está associado a múltiplos valores de variância, um para cada cenário de codificação, isto é, um conjunto {banda, resolução, tamanho de bloco, QP}.

A título de comparação, a Tabela 7 apresenta os valores de variância de corte em diferentes cenários de codificação para um *cutoff* de 0,45 e uma abordagem em cinco bandas. De modo geral, percebe-se que as variâncias de corte para a codificação de vídeos na resolução  $4096 \times 2048$  são significativamente menores do que aquelas para a codificação de vídeos na resolução  $3328 \times 1664$ . Além disso, as variâncias de corte para a região Meio-Polar são significativamente maiores do que para a região Polar, modelando o comportamento de os *skips* estarem mais acumulados nas baixas variâncias nas regiões polares. Por fim, também percebe-se que as variâncias de corte em QPs maiores são maiores, visto que ao utilizar-se QPs maiores o modo *skip* é mais comum. O Apêndice A apresenta as variâncias de corte para todos os

Tabela 7 – Variâncias de corte para *cutoff* igual a 0,45 e diferentes resoluções, tamanhos de bloco, bandas e QPs

<b>3328×1664</b>								
<b>Banda</b>	<b>Polar</b>				<b>Meio-Polar</b>			
<b>Tamanho CU</b>	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
<b>QP 22</b>	12	6	4	3	46	19	9	5
<b>QP 27</b>	41	16	7	3	85	35	14	6
<b>QP 32</b>	57	21	7	3	101	41	15	6
<b>QP 37</b>	70	23	8	4	116	45	16	6
<b>Outros QPs</b>	45	17	6	3	90	36	14	6
<b>4096×2048</b>								
<b>Banda</b>	<b>Polar</b>				<b>Meio-Polar</b>			
<b>Tamanho CU</b>	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
<b>QP 22</b>	2	2	1	1	3	2	2	1
<b>QP 27</b>	2	2	2	2	3	2	2	2
<b>QP 32</b>	2	2	2	2	5	3	3	2
<b>QP 37</b>	6	3	2	2	7	5	3	2
<b>Outros QPs</b>	3	2	2	2	4	3	2	2

conjuntos banda, resolução, tamanho de bloco, QP avaliados.

Com os pontos de operação definidos, a Figura 32 apresenta o algoritmo da técnica *Early Skip*. A técnica *Early Skip* recebe como parâmetros a **resolução** do vídeo, o **QP** sendo utilizado, o número de bandas (**nBandas**) utilizado na codificação e os valores de **cutoff**.

No algoritmo apresentado na Figura 32 a **linha 1** é a inicialização da técnica *Early Skip*, sendo assim, ela é executada logo no início da codificação. A inicialização consiste em avaliar a resolução do vídeo, o QP sendo utilizado, o número de bandas e os parâmetros de *cutoff* para as diferentes bandas, e a partir disso, calcular os limiares de variância para os quais o modo *skip* será escolhido de forma prematura. Estes limiares de variância são gerados a partir de distribuições de *skip* acumulada de acordo com a variância, como realizado na Subseção 5.2.2. Já o vetor **limiaresGerais[]** guarda os limiares de variância para os diferentes cenários de codificação que serão encontrados na codificação do vídeo atual, isto é, os limiares para diferentes bandas e tamanhos de bloco de acordo com os parâmetros recebidos.

Se três bandas estiverem sendo utilizadas, então existe apenas um valor de *cutoff* que é aplicado a ambas as bandas polares. Entretanto, se cinco bandas forem enviadas à técnica *Early Skip* então são necessários dois valores de *cutoff*: um  $cutoff_P$  para as bandas polares e um  $cutoff_{MP}$  para as bandas meio-polares. Vale ressaltar que quanto maior o valor de *cutoff*, maior será a redução de complexidade e maior o impacto na eficiência de codificação.

Além disso, embora a implementação da técnica *Early Skip* aceite apenas três

---

**Algoritmo:** Técnica *Early Skip*


---

```

Entrada: resolução, QP, nBandas, cutoffs[]

1 limiaresGerais[] = inicializa(resolução, QP, nBandas, cutoffs[])
2 para cada CU ∈ Video faça
3   | var ← calcVariância(CUAmostras)
4   | limiarAtual ← carregaLimiar(limiaresGerais[], CUTamanho, CUposY)
5   | se var ≤ limiarAtual então
6   |   | faz skip
7   | senão
8   |   | faz predição interquadros completa
9   | fim
10 fim

```

---

Figura 32 – Algoritmo da técnica *Early Skip*

ou cinco bandas, o algoritmo geral pode ser estendido para um número arbitrário de bandas desde que a análise da distribuição acumulada de modos *skip* por variância seja refeita com a respectiva distribuição de bandas.

Após a inicialização, o laço **para** da **linha 2** representa a predição interquadros de todas as CUs dentro do vídeo. A partir disso, para a **CU** que está sendo codificada atualmente duas operações sempre são realizadas. Primeiro, a variância das amostras da **CU** atual (**var**) é calculada na **linha 3**. Em seguida, na **linha 4** o cenário atual de codificação (tamanho e posição vertical da **CU** atual) é utilizado para determinar, dentre os possíveis limiares de variância salvos em **limiaresGerais[]**, qual é o limiar de variância que se aplica a **CU** sendo codificada (**limiarAtual**).

Em seguida, o par **se/senão** entre a **linha 5** e a **linha 9** caracteriza a possível decisão prematura pelo modo *skip*. Caso a variância da **CU** atual (**var**) seja menor ou igual ao **limiarAtual**, isso significa que a **CU** atual é altamente propícia a ser codificada com o modo *skip*, sendo assim, é feita uma decisão prematura pelo modo de codificação *skip* e o restante da predição interquadros para a **CU** atual é descartado. Entretanto, caso a variância (**var**) da **CU** atual seja superior ao **limiarAtual**, então não existem indícios suficientes indicando que a **CU** atual é propícia a ser codificada com o modo *skip*, e nesse caso, a predição interquadros é realizada de forma completa: o modo *merge* é avaliado, e tanto o TZS quanto a FME são realizados.

Na próxima iteração, **CU** vai representar uma região que pode ter dimensões diferentes ou estar em uma posição vertical diferente da **CU** atual. Nesse caso, na **linha 4** o **limiarAtual** é atualizado com base nestas variações para que a decisão prematura pelo modo *skip* seja avaliada com os limiares adequados.

É importante ressaltar que a técnica *Early Skip* atua somente na predição interqua-

etros, sendo assim, a predição intraquadro ainda é aplicada a CU atual independente desta ser tido codificada com o modo *skip* de forma prematura ou não. Esta etapa é de grande importância pois caso uma CU tenha sido codificada com o modo *skip* de forma prematura e esta decisão tenha resultado numa baixa eficiência de codificação, a realização da predição intraquadro pode encontrar um modo de predição mais eficiente e reduzir a penalização na eficiência de codificação. Além disso, a técnica *Early Skip* é aplicada em todos os tamanhos de CU, dessa forma, uma decisão prematura errônea num nível arbitrário da árvore de CUs pode não causar impacto algum na eficiência de codificação caso este nível já não representasse um bom candidato desde o princípio.

Esta seção descreveu o funcionamento da técnica *Early Skip*. Esta técnica consiste em avaliar a variância e a posição da CU sendo codificada, e a partir disso, tomar ou não uma decisão prematura pelo modo de predição *skip*. Como diferentes resoluções, tamanhos de bloco e QPs interferem diretamente nas ocorrências do modo de predição *skip*, a técnica *Early Skip* emprega um conjunto de limiares para se adequar às características do vídeo e parâmetros de codificação. Além disso, a técnica *Early Skip* pode ter sua agressividade controlada a partir de um parâmetro *cutoff* que escala os limiares de variância de acordo com a redução de complexidade desejada.

## 6.2 *InterTech: Reduced SR*

As avaliações realizadas na Seção 5.3 apontaram que a etapa de Busca Raster do TZS apresenta um comportamento significativamente diferente durante a codificação de vídeos convencionais e omnidirecionais. Enquanto que para vídeos convencionais a Busca Raster tem um comportamento recorrente para todas as regiões do quadro, para vídeos omnidirecionais as regiões polares tem um comportamento distinto da região central devido às distorções causadas pela projeção ERP. Além disso, os testes experimentais indicam que ao limitar-se a área de busca da Busca Raster horizontalmente, é possível reduzir a complexidade da codificação com alterações insignificantes na eficiência de codificação.

Sendo assim, a técnica *Reduced SR* consiste em reduzir horizontalmente a área de busca avaliada durante a Busca Raster de acordo com a distorção causada pela projeção ERP, de modo que onde há maior distorção a área de busca é reduzida de forma mais agressiva.

A Figura 33 descreve o algoritmo da técnica *Reduced SR*. Esta técnica recebe como entradas o número de bandas (**nBandas**) sendo utilizado, e um vetor **h[]** com o fator de escala que será utilizado para reduzir a área de busca da Busca Raster em cada banda.

A partir disso, o laço **para** na **linha 1** representa a etapa de Busca Raster sendo

---

**Algoritmo:** Técnica *Reduced SR*


---

```

Entrada: nBandas, h[]

1 para cada CU que chega na Busca Raster faça
2   | se PU  $\in$  banda[1] então
3   |   | Escala área de busca horizontal por h[1]
4   | senão se PU  $\in$  banda[2] então
5   |   | Escala área de busca horizontal por h[2]
6   |   | // Avalia se a CU está em outras bandas intermediárias...
7   | senão se PU  $\in$  banda[n] então
8   |   | Escala área de busca horizontal por h[n]
9   | fim
10  | Faz Busca Raster
11 fim

```

---

Figura 33 – Algoritmo da técnica *Reduced SR*

empregada em todas as PUs que precisam ser avaliadas por esta etapa do TZS. Na **linha 2** o condicional **PU  $\in$  banda[1]** determina se a **PU** atual pertence à **banda[1]** ou não. Caso verdadeiro (**linha 3**), então a área de busca da Busca Raster é escalada horizontalmente pelo fator **h[1]**, caso contrário (**linha 4**), então é avaliado se a **PU** pertence à outras bandas. Estes condicionais se repetem para cobrir todas as possibilidades de **nBandas** suportadas pela implementação do algoritmo, onde uma PU que pertence à **banda[n]** tem a sua área de busca escalada por **h[n]**. Uma vez que tenha sido identificado a qual banda a **PU** atual pertence e a área de busca tenha sido escalada de forma adequada, então a Busca Raster é realizada utilizando esta área de busca escalada (**linha 9**). Em todos os casos, o fator de escala **h** está dentro dos limites onde  $0 \leq h \leq 1$ , visto que um valor superior a 1 expande a área de busca.

Assim como descrito na abordagem em bandas apresentada na Figura 19, vídeos omnidirecionais na projeção ERP possuem uma distorção que tende a zero no centro do quadro, e a distorção aumenta gradualmente conforme afasta-se do centro do quadro. A partir disso, existe uma relação entre o número de bandas **nBandas** e os fatores de escala **n** que deve ser respeitada para que a técnica *Reduced SR* funcione adequadamente. Para modelar o comportamento da distorção causada pela projeção ERP o número de bandas **nBandas** enviado à técnica deve ser ímpar para dar origem a uma banda central e pelo menos um par de bandas polares complementares. Além disso, o esticamento em bandas complementares é o mesmo e na banda central ele deve tender a zero. A partir disso, o fator de escala para a banda central é sempre igual a um, isto é, a área de busca não é alterada, e a área de busca em bandas complementares deve ser limitada na mesma escala.

### 6.3 *InterTech: Reduced FME*

As avaliações conduzidas na Seção 5.4 confirmaram a **Hipótese 3** apresentada no início do Capítulo 5, isto é, o esticamento causado pela projeção ERP interfere na eficiência da FME.

De forma mais precisa, durante a projeção ERP são geradas amostras em posições horizontalmente fracionárias na superfície da esfera, que são mapeadas para posições inteiras no vídeo projetado. Desta forma, quando a FME é realizada sobre estas amostras em posições inteiras (mas que originalmente estão associadas à posições fracionárias na esfera) as amostras horizontalmente fracionárias geradas no vídeo projetado apresentam, na prática, uma precisão maior do que a esperada pelo codificador. Isso ocorre porque estão sendo geradas amostras fracionárias entre outras amostras que já representam posições originalmente fracionárias na superfície esférica.

Sendo assim, a técnica *Reduced FME* consiste em limitar a precisão empregada na FME na direção horizontal de acordo com a região do quadro sendo codificada. Como a projeção ERP não apresenta esticamento vertical, a FME é realizada normalmente na direção vertical.

Visto que o padrão HEVC emprega precisão de  $1/2$  e  $1/4$  pixel na FME, a técnica *Reduced FME* faz com que a FME funcione em um de três pontos de operação: *completo*, *moderado* ou *simplificado*. Nos três pontos de operação a FME utiliza tanto a precisão de  $1/2$  quanto  $1/4$  pixel na direção vertical, logo, as diferenças se dão apenas na direção horizontal.

No ponto de operação *completo* a FME é realizada como prevista no padrão HEVC, isto é, a FME utiliza precisão de até  $1/4$  pixel na direção horizontal. Este ponto de operação é ideal para a região central do quadro, onde o esticamento causado pela projeção ERP é suave ou até mesmo nulo. O ponto de operação *moderado* faz com que a FME utilize apenas a precisão de  $1/2$  pixel na direção horizontal. Este ponto de operação é adequado em regiões que sofrem esticamento moderado, isto é, entre o centro e as extremidades do quadro. Por fim, o ponto de operação *simplificado* descarta totalmente a FME na direção horizontal, realizando a FME apenas na direção vertical.

Além de não avaliar as amostras em algumas posições horizontalmente fracionárias, a técnica *Reduced FME* também não gera as amostras que não vão ser avaliadas posteriormente, desta forma aumentando o potencial de redução de complexidade.

Como a técnica *Reduced FME* consiste em empregar a FME com três precisões horizontais distintas, cada uma delas adequadas a um determinado nível de esticamento, a técnica *Reduced FME* trabalha numa abordagem em cinco bandas como apresentado na Figura 19, isto é, bandas polares, meio-polares e central distribuídas

---

**Algoritmo:** Técnica *Reduced FME*


---

```

1 para cada PU ∈ Video faça
2   se PU ∈ Polar então
3     | Faz FME simplificado
4   senão se PU ∈ MeioPolar então
5     | Faz FME moderado
6   senão
7     | Faz FME completo
8   fim
9 fim

```

---

Figura 34 – Algoritmo da técnica *Reduced FME*

numa proporção 15%-20%-30%-20%-15%.

A Figura 34 descreve o algoritmo da técnica *Reduced FME*. O laço **para** na **linha 1** representa a FME sendo realizada para todas as PUs do vídeo. A partir disso, cada **PU** é avaliada para, com base na sua posição vertical dentro do quadro, determinar a qual banda esta pertence. Sendo assim, na **linha 2** o condicional ***PU* ∈ *Polar*** avalia se a **PU** atual está situada na banda **Polar**. Em caso positivo, então a FME é realizada sobre a PU atual utilizando o ponto de operação *simplificado* (**linha 3**), isto é, sem gerar amostras horizontalmente fracionárias. Em caso negativo, então o condicional ***PU* ∈ *MeioPolar*** na **linha 4** avalia se a **PU** pertence a banda **Meio-Polar**. Caso este teste retorne verdadeiro, então a FME utiliza o ponto de operação *moderado* (**linha 5**) para codificar a PU atual, isto é, gera amostras horizontalmente fracionárias apenas com precisão de  $1/2$  pixel. Por fim, caso a **PU** atual não pertença nem a banda **Polar** e nem a banda **Meio-Polar** (**linha 6**), então ela deve pertencer a banda **Central**. Nesse caso, a FME é realizada no ponto de operação *completo* para a PU atual, isto é, avaliando amostras horizontalmente fracionárias com precisão de  $1/2$  e  $1/4$  pixel.

Esta seção descreveu o funcionamento da técnica de redução de complexidade *Reduced FME*. De forma geral, a técnica *Reduced FME* explora o fato da projeção ERP gerar amostras horizontalmente fracionárias por si só num momento anterior à codificação para poupar o codificador desta tarefa. Desta forma, de acordo com o esticamento horizontal nas regiões do quadro a FME alterna entre diferentes precisões horizontais visto que quanto maior o esticamento, mais as amostras geradas pela FME são redundantes com as amostras geradas pela projeção.

## 6.4 *IntraTech: Reduced Intra Sizes*

A Seção 5.5 demonstrou que durante a codificação de vídeos convencionais a distribuição de tamanhos de PU é semelhante para qualquer que seja a região do quadro sendo codificada, enquanto que durante a codificação de vídeos omnidirecionais a distribuição de tamanhos de PU varia de acordo com a região do quadro sendo codificada: enquanto que na região central de vídeos omnidirecionais a distribuição dos tamanhos de PU é similar à distribuição para vídeos convencionais, nas regiões polares as PUs maiores ( $64 \times 64$  e  $32 \times 32$ ) são escolhidas mais de 85% das vezes. Dessa forma, uma técnica que priorize a escolha destas PUs maiores nas regiões polares e descarte a avaliação das PUs menores – e consequentemente, menos prováveis – pode atingir elevadas taxas de redução de complexidade.

Entretanto, se apenas a distribuição estatística apresentada na Seção 5.5 for utilizada para gerar tal técnica de redução de complexidade é possível que a eficiência de codificação seja reduzida de forma significativa, visto que estes dados proporcionam um entendimento para fazer apenas simplificações lineares na codificação, por exemplo, nunca avaliar PUs  $4 \times 4$  nas linhas de CTU 0 e 25.

Dessa forma, além da distribuição estatística realizada anteriormente a técnica *Reduced Intra Sizes* também avalia o custo RD dos blocos previamente codificados e atual para determinar se é necessário ou não sub-particionar a CU atual e avaliar blocos menores.

O custo RD é o valor otimizado durante o RDO, isto é, o  $J$  na equação (1). Como o custo RD é uma grandeza que mede a distorção causada pela adoção de um determinado modo de codificação e a taxa de *bits* necessária para enviar estas informações, ele também pode ser considerado uma métrica de aptidão para avaliar diferentes modos de codificação, de forma que quanto menor o custo RD maior é a aptidão de um determinado modo de codificação para um bloco específico.

Para utilizar o custo RD na técnica *Reduced Intra Sizes*, a correlação temporal entre quadros vizinhos também é explorada. Como a taxa de captura de vídeos digitais costuma ser de pelo menos 24 quadros por segundo, quadros consecutivos têm características muito semelhantes, e dessa forma, métodos de codificação semelhantes costumam ser empregados em regiões co-localizadas entre estes quadros – onde métodos de codificação semelhantes dizem respeito a particionamentos similares, vetores de movimento semelhantes, modos de predição intraquadro angularmente vizinhos, etc. A partir disso, como regiões co-localizadas em quadros temporalmente próximos costumam ser codificados através de métodos de codificação semelhantes, o custo RD mínimo para codificar estas regiões co-localizadas também costuma assumir valores próximos. Sendo assim, a técnica *Reduced Intra Sizes* emprega a distribuição estatística de tamanhos de PU de acordo com linha de CTU descrita na Seção 5.5

junto da correlação entre o custo RD para codificar regiões co-localizadas em quadros temporalmente vizinhos para determinar se é necessário ou não sub-particionar a CU atual.

Visto que a técnica *Reduced Intra Sizes* explora a correlação temporal da codificação entre quadros, é necessário que exista uma referência de custo RD para cada quadro, e dessa forma, a técnica *Reduced Intra Sizes* trabalha em conjuntos de quadros denominados *Frame Sets*. O primeiro quadro em cada *Frame Set* é codificado normalmente sem nenhuma interferência, e os custos RD para este quadro são armazenados para serem utilizados como referência para o restante do *Frame Set*. A partir do segundo quadro do *Frame Set*, a técnica *Reduced Intra Sizes* utiliza os custos RD do quadro referência junto da distribuição estatística para determinar quando interromper o sub-particionamento das CUs. O número de quadros em cada *Frame Set* pode assumir um valor inteiro arbitrário entre 2 e o total de quadros da sequência, e é controlado por um parâmetro *RefreshRate*. Quanto maior for o valor de *RefreshRate*, mais quadros vão passar por redução de complexidade visto que as referências vão representar uma fração menor do total de quadros do vídeo. Entretanto, a utilização de valores de *RefreshRate* elevados também reduz a precisão do algoritmo, visto que a referência de custo RD vai estar mais afastada dos quadros ao final do *Frame Set*.

O algoritmo que descreve a técnica *Reduced Intra Sizes* é apresentado na Figura 35. A técnica *Reduced Intra Sizes* recebe três parâmetros como entrada: **RefreshRate**, **minContrib** e **th**.

O laço **para** na **linha 1** representa a predição intraquadro sendo aplicada a todas as **PUs** do vídeo. Desta forma, primeiro é realizada a predição intraquadro convencional (**linha 2**) onde os 35 modos de predição disponíveis (Figura 13) são avaliados para determinar o melhor modo para a **PU** atual, e o custo **RD** do melhor modo de predição é extraído.

Em seguida, o condicional **quadro é ref** na **da linha 3** avalia se a **PU** atual pertence ao quadro de referência do *Frame Set* ou não, com base no parâmetro **RefreshRate** utilizado. Caso a **PU** pertença ao quadro de referência, então o custo **RD** desta **PU** é salvo numa matriz de referência para o respectivo tamanho de **PU** (**linha 4**). Além disso, como o quadro de referência é codificado sem interferências, após salvar o custo **RD** a codificação segue normalmente, isto é, a **PU** é subdividida respeitando as normas do padrão HEVC (**linha 5**). Por outro lado, caso a **PU** não pertença ao quadro de referência, então é necessário avaliar outros aspectos antes de decidir subdividi-la ou interromper a subdivisão.

Se a **PU** atual não pertence ao quadro de referência, então primeiro é necessário carregar o valor de **contrib**, que representa a taxa de ocorrência do tamanho de **PU** atual na respectiva linha de CTUs com base na análise realizada na Seção 5.5. Em seguida, o custo **RD** da região co-localizada no quadro de referência (**coRD**) é calcu-

---

**Algoritmo:** Técnica *Reduced Intra Sizes*


---

**Entrada:** RefreshRate, minContrib, th

```

1 para cada PU ∈ Video faça
2   RD = Faz predição intraquadro da PU
3   se quadro é ref então
4     | Salva custo RD
5     | Subdivide PU
6   senão
7   | contrib = carrega contrib da PU
8   | coRD = calcula coRD da PU
9   | se (contrib > minContrib) && (RD < coRD × th) então
10    |   | Interrompe subdivisão
11   |   senão
12    |   | Subdivide PU
13   |   fim
14   fim
15 fim

```

---

Figura 35 – Algoritmo da técnica *Reduced Intra Sizes*

lado na **linha 8**. O custo **coRD** representa o custo **RD** associado às PUs que estão na região co-localizada à **PU** atual no quadro de referência, e mais detalhes sobre o seu cálculo são dados nos próximos parágrafos.

A partir dos valores de **contrib** e **coRD**, o condicional da **linha 9** determina se a **PU** atual deve ser subdividida ou não. Para que este condicional seja verdadeiro, então duas condições precisam ser satisfeitas: **contrib** deve ser superior a **minContrib**, e **RD** deve ser inferior a **coRD** × **th**.

A condição **contrib > minContrib** avalia se a contribuição (**contrib**) dos blocos codificados com o tamanho de **PU** atual na linha de CTUs atual é maior do que o parâmetro de contribuição mínima (**minContrib**). De forma gráfica, este condicional avalia se uma determinada barra da Figura 30 tem uma altura mínima. Dessa forma, percebe-se que **minContrib** é um limiar que pode assumir valores entre zero e um, e controla a agressividade da técnica *Reduced Intra Sizes*: valores de **minContrib** mais próximos de zero fazem com que mais PUs sejam elegíveis para a redução de complexidade, enquanto que valores de **minContrib** mais próximos de um fazem com que menos PUs sejam elegíveis para a redução de complexidade.

Já a condição **RD < coRD × th** compara o custo RD mínimo obtido na PU atual (**RD**) com o custo RD da região co-localizada (**coRD**) no quadro de referência escalado de acordo com um valor **th**. De modo geral, esta comparação avalia se o modo de predição e tamanho de bloco atuais apresentam uma boa aptidão (custo **RD** baixo)

para codificar a região atual do quadro. Sendo assim, percebe-se que o parâmetro **th** é responsável por controlar a margem de aceitação na comparação dos custos RD. Quando o valor de **th** é igual a um, testa-se se o custo RD da PU atual é menor do que o próprio custo RD da região co-localizada. Quando o valor de **th** é maior do que um, o algoritmo se torna mais agressivo visto que os testes  $RD < coRD \times th$  retornam verdadeiro com maior frequência para os mesmos valores de **RD** e **coRD**.

Sendo assim, se o condicional da **linha 9** resultar verdadeiro, então existem indícios de que o tamanho de bloco atual é ótimo para codificar esta região do vídeo: este tamanho de bloco ocorre com grande frequência nesta região do vídeo e o custo RD é baixo. Logo, é possível interromper a subdivisão em blocos menores (**linha 10**). Entretanto, se alguma das duas condições não forem satisfeitas então não existem indícios de que o tamanho de PU atual é ótimo, e neste caso, o codificador subdivide a PU novamente (**linha 12**) de acordo com o padrão HEVC.

Tendo descrito o algoritmo completo, é importante mencionar dois aspectos sobre os condicionais **contrib > minContrib** e  $RD < coRD \times th$ .

A avaliação de tamanhos de PU durante a predição intraquadro, que dá origem ao valor de limiar **contrib** para cada par linha de CTU - tamanho de bloco, foi conduzida considerando-se apenas vídeos que são codificados na resolução  $3328 \times 1664$  e apresentam 26 CTUs de altura. Entretanto, vídeos de resolução 6k e 8k devem ser codificados na resolução  $4096 \times 2048$ , logo, estes possuem 32 CTUs de altura. Para adequar a distribuição da Figura 30 – e os limiares de **contrib** – a vídeos de resolução 6k e 8k, a distribuição de vídeos 4k (que possui 26 pontos) foi interpolada pelo método de interpolação spline cúbica para gerar uma distribuição com 32 pontos, capaz de modelar a codificação de vídeos 6k e 8k. Os valores obtidos na avaliação com vídeos 4k e os valores obtidos da interpolação dos mesmos para vídeos 6k e 8k estão disponíveis no Apêndice B.

Já o custo RD da região co-localizada, utilizado no condicional  $RD < coRD \times th$  pode ser calculado de três formas distintas dependendo da diferença entre o particionamento da CTU atual e da CTU co-localizada no quadro de referência.

A Figura 36 é utilizada para descrever as três formas distintas como o custo RD da região co-localizada pode ser calculado, onde cada quadrado em linha contínua representa uma CTU, as linhas pontilhadas representam as PUs dentro da CTU, a região hachurada representa a região da PU atual e a região co-localizada no quadro de referência e os números representam o custo RD de cada PU dentro da região co-localizada.

Na Figura 36, *Atual* representa o particionamento que está sendo avaliado atualmente, isto é, uma PU  $32 \times 32$  no canto superior direito da CTU. *Caso 1* representa o caso em que a região compreendida pela PU atual compreende exatamente uma PU no quadro de referência, *Caso 2* representa o caso em que a região compreendida

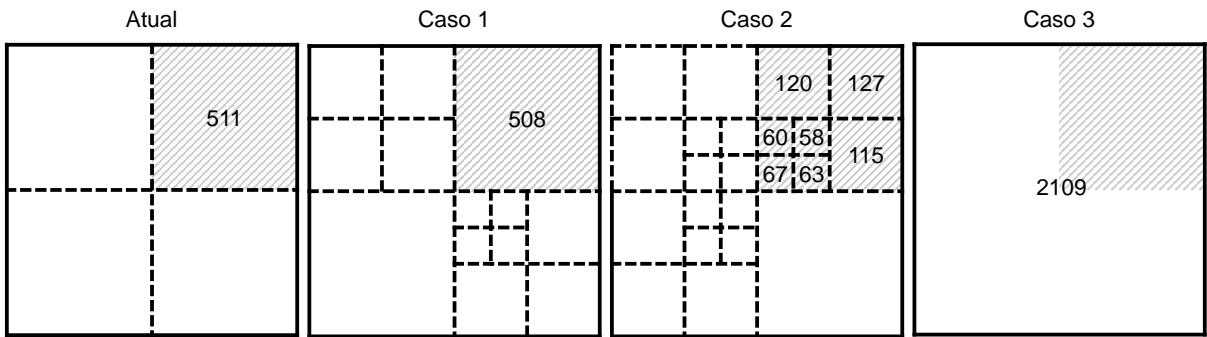


Figura 36 – Custo RD da PU atual e das PUs em três particionamentos distintos para o quadro de referência

pela PU atual compreende múltiplas PUs menores no quadro de referência e *Caso 3* representa o caso em que a região compreendida pela PU atual compreende menos do que uma PU no quadro de referência – isto é, está dentro de uma PU maior. No *Caso 3* não existem linhas pontilhadas indicando a PU pois esta tem as mesmas dimensões da CTU.

Considerando os três casos apresentados, o cálculo do custo RD para a região co-localizada em cada um dos casos se dá como descrito a seguir. Como no *Caso 1* a PU atual compreende exatamente uma PU no quadro de referência, então o custo RD da PU atual é comparada diretamente com o custo RD da PU co-localizada (testando-se  $511 < 508 \times th$ ). Já no *Caso 2* a região compreendida pela PU atual compreende múltiplas PUs no quadro de referência, dessa forma, o custo RD da PU atual é comparado com a soma dos custos RD das múltiplas PUs na região co-localizada do quadro de referência (testando-se  $511 < (120 + 127 + 60 + 58 + 67 + 63 + 115) \times th$ ). Por fim, como no *Caso 3* a região compreendida pela PU atual compreende apenas uma fração de uma PU maior no quadro de referência ( $1/4$  de uma PU  $64 \times 64$ ), então o custo RD da PU atual é comparado com uma fração do custo RD da PU que compreende a região co-localizada no quadro de referência. Como neste caso a PU atual compreende  $1/4$  da PU na região co-localizada, então o custo RD desta PU é escalado por  $1/4$  (testando-se  $511 < (2109/4) \times th$ ). Caso a PU atual compreendesse uma região de  $1/16$  ou  $1/64$  da PU no quadro de referência (quando comparando PUs  $16 \times 16$  com PUs  $64 \times 64$ , ou PUs  $8 \times 8$  com PUs  $64 \times 64$ ), então o custo RD seria escalado de acordo com estas frações.

Dessa forma, fica claro que a agressividade da redução de complexidade da técnica *Reduced Intra Sizes* é controlada pelos três parâmetros **RefreshRate**, **minContrib** e **th**. Ambos os parâmetros são definidos no momento da codificação e podem assumir qualquer valor descrito nesta seção, entretanto, uma vez definidos os parâmetros eles se mantém os mesmos até o final da codificação. Valores menores de

**minContrib** tornam a redução de complexidade mais agressiva, enquanto que valores maiores tornam a redução de complexidade mais suave. Os parâmetros **RefreshRate** e **th** têm comportamento inverso: valores maiores tornam a redução de complexidade mais agressiva, enquanto que valores menores tornam a redução de complexidade mais suave. É importante salientar que para estes três parâmetros, quanto maior for a agressividade da redução de complexidade, maior será a redução na eficiência de codificação.

Esta seção descreveu o funcionamento da técnica de redução de complexidade *Reduced Intra Sizes*. Esta técnica reduz a complexidade da predição intraquadro impedindo que algumas CUs se subdividam em CUs menores através da avaliação da distribuição estatística de tamanhos de PU por região do vídeo e dos custos RD da PU atual e da região co-localizada num quadro de referência.

## 6.5 Técnicas *InterTech* e *IntraTech*

As seções anteriores descreveram o funcionamento individual das técnicas *Early Skip*, *Reduced SR*, *Reduced FME* e *Reduced Intra Sizes*. Desta forma, esta seção busca apresentar um panorama geral de como estas técnicas funcionam em conjunto e quais são suas dependências. Além das técnicas descritas neste capítulo, esta seção também apresenta como a técnica *Reduced Intra Modes*, proveniente de um trabalho anterior (STORCH et al., 2019), trabalha em conjunto com as técnicas desenvolvidas. O diagrama que representa as interações entre estas técnicas é apresentado na Figura 37.

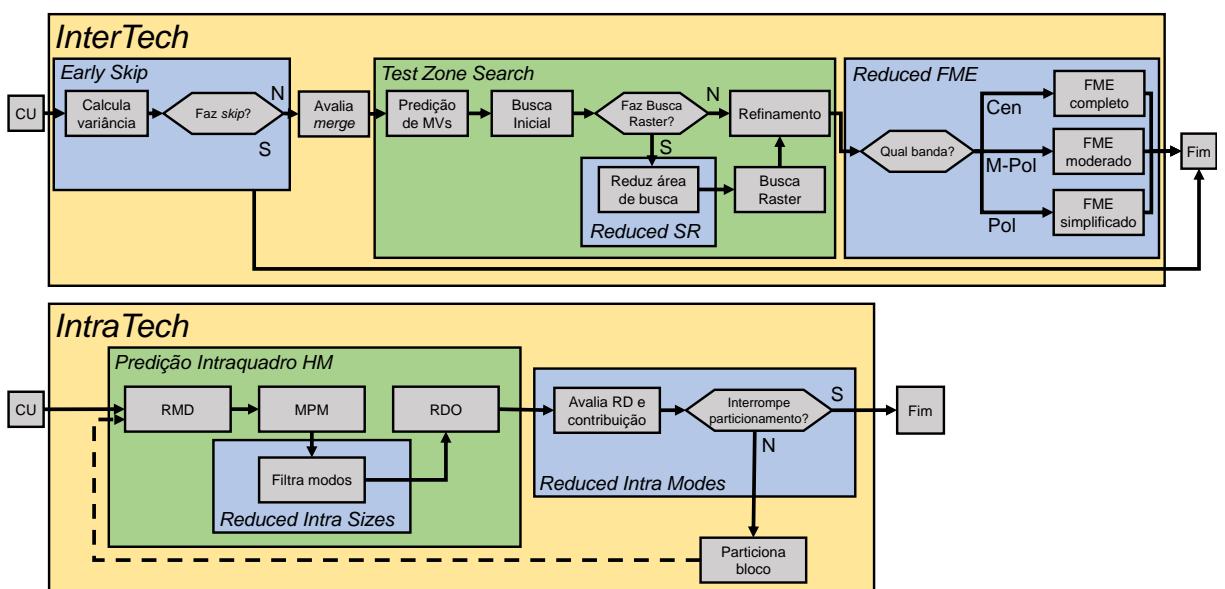


Figura 37 – Diagrama em alto nível da relação entre as técnicas desenvolvidas

A Figura 37 é composta por dois blocos principais, na cor amarela, que representam as técnicas *InterTech* e *IntraTech*. Caso o codificador utilize a configuração de codificação *Random Access*, então a técnica *InterTech* é utilizada. Por outro lado, caso o codificador utilize a configuração *All Intra* então a técnica *IntraTech* é utilizada.

Supondo que a configuração *Random Access* esteja sendo utilizada, então a primeira técnica a ser utilizada é a *Early Skip*. Esta técnica avalia a variância do bloco sendo codificado e define se este deve ser codificado com o modo *skip* de forma prematura ou não. Em caso positivo, então a CU atual é codificada com o modo *skip* e o restante da predição interquadros é descartada. Desta forma, percebe-se que caso a técnica *Early Skip* interfira na codificação, então as técnicas *Reduced SR* e *Reduced FME* não são aplicadas na CU atual visto que nem o TZS e nem a FME são executados. Entretanto, caso a técnica *Early Skip* não tome uma decisão prematura pelo modo *skip*, então a codificação segue a ordem convencional.

Após a avaliação convencional do modo *merge*, então o algoritmo *Test Zone Search*, caracterizado pelo bloco verde, é executado para encontrar um vetor de movimento inteiro para a PU sendo codificada. Primeiro o TZS executa a Predição de Vetores de Movimento e em seguida a Busca Inicial. Caso o vetor de movimento resultante da Busca Inicial seja maior que um valor *iRaster*, a Busca Raster é executada. Neste caso, então a técnica *Reduced SR* reduz as dimensões da área de busca da Busca Raster para reduzir o número de blocos candidatos avaliados de acordo com a região do quadro sendo codificado. Entretanto, caso o vetor de movimento resultante da Busca Inicial seja menor do que o valor *iRaster*, então a Busca Raster não é executada, e dessa forma, a técnica *Reduced SR* também não é utilizada.

Por fim, após a estimativa de movimento inteira através do TZS a estimativa de movimento fracionária é aplicada. Nesse caso, a técnica *Reduced FME* avalia a posição da PU sendo processada, e com base nisso, altera a precisão horizontal utilizada durante a FME. Caso a técnica *Reduced FME* detecte que o bloco atual está numa banda polar (ramo *Pol*), então a FME é realizada na configuração *simplificado*, em que amostras horizontalmente fracionárias não são avaliadas. Se for detectado que o bloco atual está numa banda meio-polar (ramo *M-Pol*), então a FME é realizada na configuração *moderado* e amostras horizontais com precisão de  $1/2$  pixel são avaliadas. Por fim, se o bloco atual estiver na banda central (ramo *Cen*), então a FME é realizada na configuração *completo*.

Por outro lado, se a configuração *All Intra* está sendo utilizada, então a primeira técnica de redução de complexidade empregada é a *Reduced Intra Modes*. A técnica *Reduced Intra Modes* fica situada dentro da *Predição Intraquadro HM*, caracterizada pelo bloco verde. A técnica *Reduced Intra Modes* avalia as decisões tomadas pelo RMD, pelo MPM e uma distribuição estatística para filtrar quais modos de predição intraquadro devem ser avaliados pelo RDO.

Em seguida, a técnica *Reduced Intra Sizes* é aplicada. Esta técnica avalia a taxa de ocorrência do tamanho de bloco atual na linha de CTUs atual, e a relação do custo RD da PU atual (gerado a partir da decisão do RDO) com a região co-localizada num quadro de referência. A partir disso, a técnica *Reduced Intra Sizes* determina se tamanhos de bloco menores devem ser testados ou se o particionamento deve ser interrompido. Caso ela decida por interromper o particionamento, então a predição é finalizada para esta região do quadro. Caso negativo, então a PU atual é particionada em PUs menores e estas são avaliadas pelas técnicas *Reduced Intra Modes* e *Reduced Intra Sizes* novamente.

Como a técnica *Reduced Intra Sizes* tem o poder de interromper a avaliação de sub-blocos do bloco atual, quando ela decide por interromper avaliações futuras ela também decide por não utilizar as técnicas *Reduced Intra Modes* e *Reduced Intra Sizes* novamente nesta região do quadro, visto que não haverão mais blocos a serem avaliados.

A conexão tracejada entre a etapa *Particiona bloco* e a *Predição Intraquadro HM* está sujeita ao tamanho de bloco sendo avaliado, visto que não é possível utilizar PUs com dimensões inferiores a  $4 \times 4$ .

Mesmo que a configuração de codificação *Random Access* utilize a predição intraquadro durante a codificação do vídeo, a estimativa de movimento é responsável por aproximadamente 70% do tempo de codificação quando a configuração *Random Access* é utilizada (PAN et al., 2016), de forma que a predição intraquadro representa uma porção marginal do tempo de codificação – os 30% restantes estão distribuídos entre predição intraquadro, transformadas, quantização, codificação de entropia, etc. Um dos motivos disso ocorrer é porque, diferente da configuração *All Intra* que aplica a predição intraquadro a todos os blocos do vídeo, a configuração *Random Access* possui heurísticas que fazem com que o codificador utilize a predição intraquadro apenas numa fração dos blocos. Avaliações preliminares utilizando a configuração *Random Access* junto da técnica *IntraTech* indicaram que a redução de complexidade obtida pela técnica *IntraTech* é marginal nesta configuração de codificação. Isso ocorre porque parte da extração de dados e do processamento são realizados mesmo quando a predição intraquadro não é realizada, e este *overhead* pode ultrapassar a redução de complexidade obtida nas poucas vezes em que a predição intraquadro é utilizada. Desta forma, a técnica *IntraTech* não é utilizada em conjunto com a configuração *Random Access*.

## 7 RESULTADOS EXPERIMENTAIS

Este capítulo apresenta a avaliação das técnicas de redução de complexidade desenvolvidas no decorrer deste trabalho. Para isso, as técnicas de redução de complexidade intraquadro são avaliadas utilizando-se a configuração de codificação *All Intra*, enquanto que as técnicas de redução de complexidade interquadros são avaliadas com a configuração de codificação *Random Access*.

As doze sequências recomendadas pelas CTCs360 (BOYCE et al., 2017) foram codificadas para obter os resultados experimentais, e seus nomes e respectivas resoluções são apresentadas nas colunas *Vídeo* e *Resolução original* na Tabela 8. Este conjunto é composto por duas sequências 4k, quatro sequências 6k e seis sequências 8k. O primeiro quadro de cada uma destas sequências é apresentado no Anexo A. Além disso, como descrito na Seção 5.1, as CTCs360 recomendam que vídeos 4k sejam codificados na resolução  $3328 \times 1664$ , enquanto que vídeos 6k e 8k sejam codificados na resolução  $4096 \times 2048$ . Sendo assim, a resolução em que cada vídeo é codificado, como recomendado pelas CTCs360, é apresentada na coluna *Resolução de codificação*. Por fim, as sequências indicadas como *Sequência<sup>A</sup>* foram utilizadas tanto na obtenção dos resultados quanto nas avaliações conduzidas no Capítulo 5.

Tabela 8 – Sequências usadas para a obtenção dos resultados experimentais

<b>Vídeos</b>	<b>Resolução original</b>	<b>Resolução de codificação</b>
<i>AerialCity<sup>A</sup></i>	$3840 \times 1920$	$3328 \times 1664$
<i>PoleVault<sup>A</sup></i>	$3840 \times 1920$	$3328 \times 1664$
<i>Balboa</i>	$6144 \times 3072$	$4096 \times 2048$
<i>BranCastle</i>	$6144 \times 3072$	$4096 \times 2048$
<i>Broadway<sup>A</sup></i>	$6144 \times 3072$	$4096 \times 2048$
<i>Landing</i>	$6144 \times 3072$	$4096 \times 2048$
<i>ChairliftRide</i>	$8192 \times 4096$	$4096 \times 2048$
<i>GasLamp</i>	$8192 \times 4096$	$4096 \times 2048$
<i>Harbor</i>	$8192 \times 4096$	$4096 \times 2048$
<i>KiteFlite</i>	$8192 \times 4096$	$4096 \times 2048$
<i>SkateboardInlot<sup>A</sup></i>	$8192 \times 4096$	$4096 \times 2048$
<i>Trolley</i>	$8192 \times 4096$	$4096 \times 2048$

Devido ao extenso conjunto de técnicas desenvolvidas e os longos tempos de processamento necessário para codificar vídeos omnidirecionais, apenas os 100 primeiros quadros de cada vídeo foram codificados. Os demais aspectos da codificação foram guiados pelas CTCs360 (BOYCE et al., 2017).

Para obter os resultados experimentais, dois servidores semelhantes foram utilizados. O primeiro deles, que será denominado **G5**, é equipado com um processador Intel Xeon Gold 5118 @ 2,30GHz e 56GB de memória RAM. O segundo servidor será denominado **G6**, e é equipado com um processador Intel Xeon Gold 5120 @ 2,20GHz 64GB de memória RAM. Mesmo com servidores semelhantes, a metodologia de avaliação das técnicas desenvolvidas não compara os resultados entre os dois servidores: o conjunto de sequências de teste *AerialCity*, *PoleVault*, *Harbor*, *KiteFlite*, *SkateboardInLot* e *Trolley* foram codificadas no servidor **G6**, enquanto que as sequências *Balboa*, *BranCastle*, *ChairliftRide*, *GasLamp*, *Broadway* e *Landing* foram codificadas no servidor **G5**. Desta forma, o resultado de todas as técnicas aplicadas a uma mesma sequência são provenientes do mesmo *hardware* e é possível medir o desempenho das técnicas. Além disso, mesmo tendo múltiplos núcleos de processamento disponíveis, apenas uma codificação foi realizada por vez em cada servidor para evitar ruído entre os processos e competição por recursos.

Como este trabalho propõe um conjunto de técnicas de redução de complexidade para a codificação de vídeos omnidirecionais, um dos aspectos avaliados é a própria redução de complexidade. Esta avaliação considerou o tempo de processamento necessário para codificar cada sequência. Além disso, a eficiência de codificação foi avaliada através da métrica BD-BR (BJØNTEGAARD, 2001), descrita em detalhes na Seção 5.1.2. Por fim, a Redução de Complexidade Relativa (RCR) também foi avaliada. A RCR representa qual seria a redução de complexidade para cada 1% de BD-BR causado por uma técnica, para que seja possível avaliar a redução de complexidade e a eficiência de codificação simultaneamente. A partir disso, percebe-se que quanto maior for a RCR de uma técnica, melhor é o seu desempenho. Além disso, caso uma técnica apresente BD-BR igual a zero, então a RCR vai tender a infinito, o que significa que a técnica é capaz de atingir redução de complexidade sem comprometer em nada a eficiência de codificação. Entretanto, caso uma técnica gere um aumento na eficiência de codificação (BD-BR negativo) a RCR assume um valor negativo e perde seu significado. Isso ocorre porque deseja-se que ambos os valores tenham a maior magnitude possível, dando origem a uma RCR de -1. Entretanto, quanto mais negativo for a RCR, maior é a redução de complexidade em relação ao aumento de eficiência de codificação; e quanto mais próximo de 0 for a RCR, maior é o aumento de eficiência de codificação em relação à redução de complexidade. Em todas as avaliações experimentais, valores negativos e infinitos de RCR são desconsiderados durante o cálculo de RCR médio.

As próximas seções deste capítulo se organizam da seguinte forma. A Seção 7.1 apresenta os resultados da técnica *Early Skip*, a Seção 7.2 apresenta os resultados da técnica *Reduced SR* e a Seção 7.3 apresenta os resultados da técnica *Reduced FME*. Em seguida, a Seção 7.4 apresenta os resultados obtidos quando todas as técnicas *InterTech* estão ativadas simultaneamente, isto é, *Early Skip*, *Reduced SR* e *Reduced FME* trabalhando juntas. A seguir a Seção 7.5 apresenta os resultados da técnica *Reduced Intra Sizes*, enquanto que a Seção 7.6 apresenta os resultados obtidos quando todas as técnicas *IntraTech* são utilizadas simultaneamente, isto é, tanto *Reduced Intra Sizes* quanto *Reduced Intra Modes* (STORCH et al., 2019).

## 7.1 Resultados de *Early Skip*

Esta seção apresenta os resultados de redução de complexidade e eficiência de codificação obtidos pela técnica *Early Skip*. Para que seja possível avaliar os resultados desta técnica de forma isolada, nenhuma das demais técnicas de redução de complexidade foi utilizada simultaneamente.

Além disso, os resultados apresentados nesta seção correspondem à utilização da técnica *Early Skip* com uma abordagem em cinco bandas, como apresentado na Figura 19. Como cinco bandas são utilizadas, então a técnica *Early Skip* atua em somente 70% do vídeo, visto que a banda central corresponde a 30% do quadro e é codificada de forma convencional.

Além disso, devido à utilização de cinco bandas existem dois pares de bandas complementares (polares e meio-polares) e uma banda central, sendo assim, a técnica *Early Skip* é controlada por dois parâmetros de *cutoff*, um  $cutoff_P$  e um  $cutoff_{MP}$ . O parâmetro  $cutoff_P$  é responsável por determinar a variância de corte para os blocos localizados nas bandas polares, enquanto que o parâmetro  $cutoff_{MP}$  é responsável por determinar a variância de corte para os blocos localizados nas bandas meio-polares.

Diversas combinações de  $cutoff_P$  e  $cutoff_{MP}$  foram avaliadas com o objetivo de encontrar o conjunto de parâmetros que leva aos melhores resultados, e a combinação que obteve o melhor custo-benefício em relação a redução de complexidade e eficiência de codificação foi utilizar  $cutoff_P$  igual a 0,35 e  $cutoff_{MP}$  igual a 0,15. Sendo assim, os resultados para a técnica *Early Skip* com os parâmetros de *cutoff* recém mencionados são apresentados na Tabela 9, onde a coluna RC representa os resultados de redução de complexidade, a coluna BD-BR representa os resultados de eficiência de codificação e a coluna RCR representa a redução de complexidade relativa. Os valores infinitos e negativos de RCR foram desconsiderados no cálculo de RCR médio.

Ao analisar os dados da Tabela 9 percebe-se que a técnica *Early Skip* é capaz de atingir uma redução de complexidade significativa com um impacto muito pequeno na

Tabela 9 – Redução de complexidade e variação de eficiência de codificação da técnica *Early Skip*

Resolução	Vídeo	RC	BD-BR	RCR
4k	<i>AerialCity</i>	9,99%	0,16%	61,24
	<i>PoleVault</i>	10,09%	0,03%	401,72
6k	<i>Balboa</i>	21,44%	0,30%	70,38
	<i>BranCastle</i>	2,67%	0,34%	7,78
	<i>Broadway</i>	20,61%	0,66%	31,43
8k	<i>Landing</i>	12,49%	0,53%	23,79
	<i>ChairliftRide</i>	5,43%	0,09%	62,72
	<i>GasLamp</i>	19,61%	0,00%	Inf
8k	<i>Harbor</i>	21,16%	0,06%	380,66
	<i>KiteFlite</i>	7,58%	-0,01%	-1162,04
	<i>SkateboardInlot</i>	13,47%	0,45%	29,94
	<i>Trolley</i>	18,58%	0,00%	Inf
<b>Média 4k</b>		<b>10,04%</b>	<b>0,09%</b>	<b>231,48</b>
<b>Média 6k</b>		<b>14,30%</b>	<b>0,46%</b>	<b>33,34</b>
<b>Média 8k</b>		<b>14,31%</b>	<b>0,10%</b>	<b>157,77</b>
<b>Média geral</b>		<b>13,59%</b>	<b>0,24%</b>	<b>118,85</b>

eficiência de codificação.

Quando os resultados de redução de complexidade são avaliados, percebe-se que o melhor caso ocorreu durante a codificação da sequência *Balboa*, onde a técnica *Early Skip* atingiu 21,44% de redução de complexidade. Já o pior caso ocorreu durante a codificação da sequência *BranCastle*, onde uma redução de 2,64% de complexidade foi observada. Em média, a técnica *Early Skip* atingiu 10,04%, 14,30% e 14,31% de redução de complexidade para vídeos 4k, 6k e 8k, respectivamente, enquanto que a redução de complexidade média para todos os vídeos foi de 13,59%.

A variabilidade na redução de complexidade observada na Tabela 9 está associada à homogeneidade das sequências avaliadas nas bandas polares e meio-polares. Em sequências como *Balboa* e *Harbor*, onde existe uma grande redundância nos blocos das bandas polares e meio-polares, a técnica *Early Skip* é capaz de detectar tais homogeneidades e optar por fazer uma decisão prematura pelo modo *skip* com uma elevada frequência, acarretando grandes reduções de complexidade. Por outro lado, sequências como *BranCastle* e *ChairliftRide* apresentam texturas mais detalhadas nas regiões de atuação da técnica *Early Skip*, de modo que a decisão prematura pelo modo *skip* não ocorre com frequência.

Além disso, os resultados médios mostram que a técnica *Early Skip* obteve reduções de complexidade mais significativas nas sequências de resolução 6k e 8k. Isso pode estar relacionado ao fato de que, como recomendado pelas CTCs360 (BOYCE et al., 2017), sequências 6k e 8k são ambas redimensionadas para a resolução

$4096 \times 2048$  antes da codificação, enquanto que sequências 4k são redimensionadas para a resolução  $3328 \times 1664$ . Como discutido anteriormente, vídeos em resoluções maiores tendem a apresentar blocos naturalmente mais homogêneos devido à quantidade de informação por bloco, tornando estes mais suscetíveis a serem codificados com o modo *skip* de forma prematura – mesmo com limiares de variância distintos para resoluções  $4096 \times 2048$  e  $3328 \times 1664$ .

A coluna BD-BR mostra que a técnica *Early Skip* causou variações pequenas na eficiência de codificação. Em relação a eficiência de codificação, a técnica *Early Skip* obteve os melhores resultados durante a codificação da sequência *KiteFlite*, onde foi possível aumentar a eficiência de codificação do codificador mesmo reduzindo a complexidade deste, e durante a codificação das sequências *GasLamp* e *Trolley*, onde a eficiência de codificação não foi alterada. Por outro lado, os piores resultados ocorreram durante a codificação das sequências *Broadway* e *Landing*, onde valores de BD-BR iguais a 0,66% e 0,53% são encontrados, respectivamente. Em média, a técnica *Early Skip* acarretou valores de BD-BR iguais a 0,09%, 0,46% e 0,10% para sequências de resolução 4k, 6k e 8k, respectivamente, enquanto que o BD-BR médio para todos os vídeos é igual a 0,24%.

Embora seja incomum que técnicas de redução de complexidade sejam capazes de aumentar a eficiência de codificação, a técnica *Early Skip* atingiu um aumento na eficiência de codificação em um dos vídeos. Isso ocorre porque o modo *skip* apresenta uma taxa de compressão significativamente maior do que os demais modos de predição interquadros, como vetores de movimento convencionais e o modo *merge*, devido ao descarte dos resíduos de predição. Sendo assim, em alguns casos a redução no *bitrate* necessário para representar o vídeo é maior do que a degradação visual, e dessa forma é possível aumentar a eficiência de codificação. Além destes casos onde houve um aumento na eficiência de codificação, múltiplas sequências apresentaram um BD-BR abaixo de 0,10%, o que indica que a técnica *Early Skip* obteve uma alta taxa de acerto quando realizou decisões prematuras pelo modo *skip*.

Quando a redução de complexidade relativa é analisada, percebe-se que existem valores muito elevados e também negativos para algumas sequências. Estes casos ocorrem porque como o BD-BR é muito próximo de zero nas respectivas sequências, a redução de complexidade relativa tende a infinito. Os melhores casos no aspecto CRC ocorreram para as sequências *GasLamp*, *KiteFlite* e *Trolley*, onde foi possível atingir elevadas reduções de complexidade com variações desprezíveis de eficiência de codificação. Além disso, o pior caso no aspecto RCR ocorreu durante a codificação da sequência *BranCastle*, que obteve uma RCR igual a 7,78. Isso ocorre porque a menor redução de complexidade da técnica *Early Skip* se deu nesta sequência, e uma das piores reduções na eficiência de codificação se deu na mesma sequência. Isso mostra que a técnica *Early Skip* realizou poucas decisões prematuras pelo modo

*skip*, e muitas destas decisões foram errôneas quando comparadas às decisões do codificador original.

Por fim, é possível perceber que a técnica *Early Skip* é capaz de atingir reduções de complexidade significativas com pequenos impactos na eficiência de codificação na maioria dos casos. Além disso, em alguns casos a técnica *Early Skip* é capaz de aumentar a eficiência de codificação enquanto reduz a complexidade do codificador.

## 7.2 Resultados de *Reduced SR*

Esta seção apresenta os resultados obtidos quando a técnica *Reduced SR* é empregada para acelerar a codificação de vídeos omnidirecionais. Assim como no caso anterior, as demais técnicas de redução de complexidade foram desativas para avaliar a performance individual da técnica *Reduced SR*.

A técnica *Reduced SR* pode trabalhar com um conjunto arbitrário (desde que ímpar) de bandas, entretanto, os resultados apresentados nesta seção foram obtidos com uma abordagem em cinco bandas, como apresentado na Figura 19. Como foi utilizada uma abordagem em cinco bandas, existem dois fatores de escala, um para as bandas polares e outro para as bandas meio-polares. Diversas combinações destes dois parâmetros foram avaliadas, e constatou-se que reduzir a área de busca das bandas polares para 30% de sua largura original leva aos melhores resultados, enquanto que reduzir a área de busca das bandas meio-polares para 70% de sua largura original leva aos melhores resultados. Sendo assim, estes foram os fatores de escala utilizados para obter os resultados experimentais. Além disso, como a abordagem em cinco bandas foi utilizada a técnica *Reduced SR* atua somente em 70% do vídeo visto que a banda central (correspondente a 30% do vídeo) é codificada normalmente.

Os resultados obtidos pela técnica *Reduced SR* com os parâmetros recém citados são apresentados na Tabela 10, onde a coluna RC representa a redução de complexidade obtida pela técnica, a coluna BD-BR representa a variação na eficiência de codificação e a coluna RCR representa a redução de complexidade relativa obtida pela técnica. Valores negativos e infinitos de RCR foram desconsiderados no cálculo de RCR médio.

Quando os resultados de redução de complexidade da Tabela 10 são analisados percebe-se que a técnica *Reduced SR* atinge reduções de complexidade modestas, entre 0,09% e 5,64%. No melhor dos casos, a técnica reduziu a complexidade da codificação em 5,64% para o vídeo *BranCastle*, enquanto que o pior caso se deu no vídeo *Trolley*. Em média, a técnica *Reduced SR* obteve reduções de complexidade iguais a 0,72%, 4,14% e 1,62% para vídeos 4k, 6k e 8k, respectivamente, enquanto que a redução de complexidade média geral foi de 2,31%.

Estes resultados modestos de redução de complexidade estão relacionados ao

Tabela 10 – Redução de complexidade e variação de eficiência de codificação da técnica *Reduced SR*

Resolução	Vídeo	RC	BD-BR	RCR
<b>4k</b>	<i>AerialCity</i>	0,59%	0,00%	-700,47
	<i>PoleVault</i>	0,84%	0,00%	429,08
<b>6k</b>	<i>Balboa</i>	3,02%	0,03%	89,82
	<i>BranCastle</i>	5,64%	0,03%	217,83
	<i>Broadway</i>	3,24%	0,04%	84,99
<b>8k</b>	<i>Landing</i>	4,68%	-0,03%	-137,15
	<i>ChairliftRide</i>	3,78%	-0,02%	-159,26
	<i>GasLamp</i>	0,28%	0,00%	256,18
<b>Média</b>	<i>Harbor</i>	0,17%	0,03%	4,76
	<i>KiteFlite</i>	0,70%	-0,02%	-42,80
	<i>SkateboardInlot</i>	4,72%	-0,04%	-114,80
	<i>Trolley</i>	0,09%	0,00%	-49,23
<b>4k</b>	<b>Média 4k</b>	<b>0,72%</b>	<b>0,00%</b>	<b>429,08</b>
<b>6k</b>	<b>Média 6k</b>	<b>4,14%</b>	<b>0,02%</b>	<b>130,88</b>
<b>8k</b>	<b>Média 8k</b>	<b>1,62%</b>	<b>-0,01%</b>	<b>130,47</b>
	<b>Média geral</b>	<b>2,31%</b>	<b>0,00%</b>	<b>180,44</b>

comportamento iterativo do algoritmo TZS (ver Seção 3.3.3). Como a técnica *Reduced SR* reduz a área de busca da etapa de Busca Raster, o vetor de movimento proveniente desta etapa é sub-ótimo, e desta forma, é possível que a etapa de Refinamento precise fazer mais iterações até encontrar uma condição de parada. Além disso, alterar a área de busca da Busca Raster pode interferir no vetor de movimento final escolhido para o bloco sendo codificado, o que afeta a etapa de Predição de Vetores de Movimento dos blocos subsequentes, e por fim pode fazer com que mais blocos passem pela etapa de Busca Raster, atenuando a redução de complexidade obtida pela redução da área de busca.

Já quando os valores de BD-BR da técnica *Reduced SR* são avaliados, percebe-se que a técnica causa variações desprezíveis na eficiência de codificação. No melhor caso, a técnica *Reduced SR* aumentou a eficiência de codificação, atingindo um BD-BR igual a -0,04% na sequência *SkateboardInLot*. Já no pior caso, a técnica *Reduced SR* causou uma redução na eficiência de codificação, com um BD-BR igual a 0,04% durante a codificação de *Broadway*. Por fim, o BD-BR médio da técnica *Reduced SR* para vídeos de resolução 4k, 6k e 8k foi 0,00%, 0,02% e -0,01%, enquanto que o BD-BR médio para todas as sequências foi 0,00%.

Os casos onde a técnica *Reduced SR* obteve valores de BD-BR negativos – um aumento na eficiência de codificação – representam casos onde a redução da área de busca da Busca Raster tornou a distribuição de vetores de movimento mais homogênea, de forma que o aumento na taxa de compressão atingida pela codificação de

entropia superou a degradação de qualidade causada por vetores de movimento sub-ótimos. Já os casos onde o BD-BR foi positivo representam os casos onde a utilização de vetores movimento sub-ótimos causou uma degradação na qualidade significativa, enquanto que a maior homogeneidade dos vetores de movimento não levou a um aumento significativo na taxa de compressão.

Por fim, como os valores de BD-BR estiveram próximos de zero em todos os casos, uma análise da coluna RCR aponta altos valores de redução de complexidade relativa. Todos os casos onde a RCR é negativa podem ser considerados melhores casos, visto que aumentaram a eficiência de codificação ao mesmo tempo em que reduziram a complexidade da codificação. Já o pior caso ocorreu na sequência *Harbor*, onde a redução de complexidade relativa foi igual a 4,76.

### 7.3 Resultados de *Reduced FME*

Esta seção apresenta os resultados experimentais obtidos pela técnica *Reduced FME*. Assim como anteriormente, as demais técnicas de redução de complexidade foram desativadas para avaliar a técnica *Reduced FME* de forma isolada. Além disso, a técnica *Reduced FME* trabalha com uma abordagem em cinco bandas como apresentado na Figura 19, sendo que nas bandas polares a FME não é realizada na direção horizontal, nas bandas meio-polares a FME é realizada na horizontal apenas com precisão de  $1/2$  pixel, e na banda central a FME é realizada de forma completa. Desta forma, a técnica *Reduced FME* atua em somente 70% do quadro, visto que a banda central corresponde a 30% do quadro.

Os resultados da técnica *Reduced FME* são apresentados na Tabela 11, onde a coluna RC representa a redução de complexidade, a coluna BD-BR representa a variação na eficiência de codificação e a coluna RCR representa a redução de complexidade relativa.

Ao avaliar a Tabela 11 percebe-se que a técnica *Reduced FME* obteve redução de complexidade de complexidade com um pequeno impacto na eficiência de codificação na maioria dos casos.

Em relação à redução de complexidade, o melhor caso ocorreu para a codificação da sequência *AerialCity* onde a técnica *Reduced FME* alcançou 12,61% de redução de complexidade, enquanto que o pior caso se deu na sequência *BranCastle*, onde a técnica *Reduced FME* obteve 6,79% de redução de complexidade. Em média, a técnica *Reduced FME* obteve redução de complexidade igual a 11,88%, 8,13% e 11,02% para sequências de resolução 4k, 6k e 8k, respectivamente, enquanto que a redução de complexidade média para todas as sequências foi igual a 10,20%.

É importante mencionar que durante uma codificação sem técnicas de redução de complexidade, a FME apresenta complexidade constante para vídeos de mesma re-

Tabela 11 – Redução de complexidade e variação de eficiência de codificação da técnica *Reduced FME*

Resolução	Vídeo	RC	BD-BR	RCR
4k	<i>AerialCity</i>	12,61%	0,65%	19,29
	<i>PoleVault</i>	11,14%	0,29%	38,94
6k	<i>Balboa</i>	9,57%	0,52%	18,27
	<i>BranCastle</i>	6,79%	0,61%	11,08
	<i>Broadway</i>	8,42%	1,00%	8,46
8k	<i>Landing</i>	7,73%	0,57%	13,48
	<i>ChairliftRide</i>	10,15%	0,93%	10,91
	<i>GasLamp</i>	12,29%	0,04%	283,50
	<i>Harbor</i>	12,01%	0,10%	121,93
	<i>KiteFlite</i>	11,23%	0,27%	41,04
	<i>SkateboardInlot</i>	8,04%	0,05%	160,54
	<i>Trolley</i>	12,41%	0,01%	883,28
<b>Média 4k</b>		<b>11,88%</b>	<b>0,47%</b>	<b>29,11</b>
<b>Média 6k</b>		<b>8,13%</b>	<b>0,68%</b>	<b>12,22</b>
<b>Média 8k</b>		<b>11,02%</b>	<b>0,24%</b>	<b>250,20</b>
<b>Média geral</b>		<b>10,20%</b>	<b>0,42%</b>	<b>134,23</b>

solução visto que realiza o mesmo conjunto de cálculos para todos os blocos. Além disso, a técnica *Reduced FME* é realizada da mesma forma para todos os blocos numa mesma banda, de forma que a redução de complexidade também é constante. Sendo assim, a variabilidade nos valores de redução de complexidade presentes na Tabela 11 se dá pela variabilidade presente em outros momentos da codificação, isto é, quanto maior for a complexidade das outras etapas da codificação (que não apresentam complexidade constante), menor será a contribuição de complexidade da FME, e desta forma menor será a redução de complexidade geral obtida pela técnica *Reduced FME*.

Em relação à eficiência de codificação, percebe-se que também há uma grande variação nos valores de BD-BR. No melhor caso, a técnica *Reduced FME* acarretou um BD-BR igual a 0,01% na sequência *Trolley*. Um valor desta magnitude é considerado desprezível, e dessa forma a técnica *Reduced FME* alcançou redução de complexidade sem degradar a eficiência de codificação. Por outro lado, quando a técnica *Reduced FME* foi aplicada à sequência *Broadway* um BD-BR igual a 1,00% foi observado. Este valor indica que a técnica *Reduced FME* simplificou a codificação em regiões do vídeo sensíveis e acarretou uma redução visível na eficiência de codificação. De modo geral, a técnica *Reduced FME* levou a valores de BD-BR iguais a 0,47%, 0,68% e 0,24% para sequências de resolução 4k, 6k e 8k, respectivamente, enquanto que o valor de BD-BR médio para todas as sequências é igual a 0,42%.

A variabilidade nos valores de BD-BR na Tabela 11 deveu-se às características de

textura e movimento de cada sequência. Para algumas sequências, como *Trolley* e *GasLamp*, a FME traz um ganho marginal de qualidade para a sequência e a maior variabilidade nos vetores de movimento devido a granularidade da FME demanda um *bitrate* maior para representar o vídeo codificado. Dessa forma, ao limitar-se as possibilidades de vetores de movimento resultantes da FME a degradação na qualidade é pequena, mas a maior homogeneidade dos vetores de movimento demanda um *bitrate* menor, acarretando um impacto marginal na eficiência de codificação. Em outras sequências, como *Broadway* e *ChairliftRide*, a FME traz um ganho significativo de qualidade para a sequência com um impacto similar no *bitrate*, fazendo com que a limitação do funcionamento da FME cause uma degradação visual significativa no vídeo com uma redução de *bitrate* pequena, acarretando uma redução de eficiência de codificação moderada.

Quando avalia-se a redução de complexidade relativa da técnica *Reduced FME*, o melhor caso ocorreu durante a codificação da sequência *Trolley* que obteve RCR superior a 883, e o pior caso se deu na codificação da sequência *Broadway*, onde um RCR igual a 8,46 é observado.

De modo geral, percebe-se que a técnica *Reduced FME* é capaz de reduzir a complexidade com um pequeno impacto na eficiência de codificação, o que indica que ela é uma candidata a ser utilizada em sistemas de codificação de vídeos omnidirecionais que demandem uma redução de complexidade e possam assumir pequenas degradações na eficiência de codificação.

## 7.4 Resultados das técnicas *InterTech* integradas

Esta seção apresenta os resultados obtidos com as técnicas *InterTech* integradas, isto é, quando *Early Skip*, *Reduced SR* e *Reduced FME* estão ativadas simultaneamente. Estas três técnicas foram utilizadas com o mesmo conjunto de parâmetros empregado nas avaliações individuais, sendo assim, a técnica *InterTech* atua sobre 70% das amostras do quadro e os 30% correspondentes às amostras da banda central são codificados sem interferência.

Os resultados da técnica *InterTech* são apresentados na Tabela 12, onde a coluna RC representa a redução de complexidade, a coluna BD-BR representa a variação na eficiência de codificação e a coluna RCR representa a redução de complexidade relativa.

Quando os resultados de redução de complexidade da técnica *InterTech* são analisados, percebe-se que no melhor caso ela é capaz de atingir 28,57% de redução de complexidade para a sequência *Harbor*, enquanto que o pior caso ocorreu para a sequência *ChairliftRide*, onde uma redução de complexidade igual a 17,82% é observada. Em média, a redução de complexidade atingida pela técnica *InterTech* é igual a

Tabela 12 – Redução de complexidade e variação de eficiência de codificação da técnica *InterTech*

Resolução	Vídeo	RC	BD-BR	RCR
4k	<i>AerialCity</i>	20,81%	0,79%	26,38
	<i>PoleVault</i>	19,71%	0,31%	63,71
6k	<i>Balboa</i>	28,39%	0,85%	33,53
	<i>BranCastle</i>	14,54%	0,89%	16,28
	<i>Broadway</i>	27,63%	1,75%	15,76
8k	<i>Landing</i>	22,01%	1,13%	19,51
	<i>ChairliftRide</i>	17,82%	1,09%	16,42
	<i>GasLamp</i>	26,78%	0,05%	590,33
8k	<i>Harbor</i>	28,57%	0,14%	201,43
	<i>KiteFlite</i>	17,96%	0,30%	59,68
	<i>SkateboardInlot</i>	23,06%	0,51%	45,34
	<i>Trolley</i>	26,78%	0,02%	1231,12
<b>Média 4k</b>		<b>20,26%</b>	<b>0,55%</b>	<b>45,05</b>
<b>Média 6k</b>		<b>23,14%</b>	<b>1,16%</b>	<b>21,27</b>
<b>Média 8k</b>		<b>23,50%</b>	<b>0,35%</b>	<b>357,39</b>
<b>Média geral</b>		<b>22,84%</b>	<b>0,65%</b>	<b>193,29</b>

20,26%, 23,14% e 23,50% para sequências de resolução 4k, 6k e 8k, enquanto que a redução de complexidade média para todas as sequências é igual a 22,84%.

Também percebe-se que estes resultados de redução de complexidade para a técnica *InterTech* são menores do que a soma das reduções de complexidade das três técnicas funcionando de forma individual. Isso ocorre porque como descrito na Seção 6.5, as decisões de algumas técnicas influenciam no funcionamento das demais. Para o caso da técnica *InterTech* mais precisamente, sempre que a técnica *Early Skip* realiza uma decisão prematura pelo modo *skip*, as técnicas *Reduced SR* e *Reduced FME* não são utilizadas.

Quando os dados de BD-BR são analisados, percebe-se que nos melhores casos a técnica *InterTech* acarretou um BD-BR igual a 0,02% e 0,05% nos vídeos *Trolley* e *GasLamp*, respectivamente. Enquanto isso, os piores casos ocorreram nas sequências *Broadway* e *Landing* onde o valor de BD-BR foi igual a 1,75% e 1,13%, respectivamente. Em média, a técnica *InterTech* causou um BD-BR igual a 0,55%, 1,16% e 0,35% para vídeos de resolução 4k, 6k e 8k, respectivamente, enquanto que o BD-BR médio para todos os vídeos é igual a 0,65%.

Quando estes resultados de BD-BR são comparados ao BD-BR individual das três técnicas que compõem a técnica *InterTech*, percebe-se que em alguns casos o BD-BR da técnica *InterTech* é superior à soma do BD-BR das técnicas constituintes, enquanto que em outros casos ele é menor.

Os casos onde o BD-BR da técnica *InterTech* é superior à soma do BD-BR das

suas técnicas constituintes estão relacionados com o fato de que quando apenas uma das técnicas constituintes é utilizada, o impacto negativo na eficiência de codificação causado por esta técnica pode ser atenuado pelas decisões das demais etapas da codificação, reduzindo o impacto na eficiência de codificação. Por exemplo, caso a limitação na área de busca causada pela técnica *Reduced SR* tenha dado origem a um vetor de movimento sub-ótimo, é possível que durante a FME este vetor de movimento seja refinado e atinja uma maior eficiência de codificação. Entretanto, quando as três técnicas são utilizadas simultaneamente na técnica *InterTech*, o impacto negativo que uma técnica causa na eficiência de codificação pode não ser atenuado por outras etapas da codificação pois estas outras também foram simplificadas, e causam um impacto ainda maior na eficiência de codificação. Um exemplo deste caso ocorre quando a técnica *Reduced SR* dá origem a um vetor de movimento sub-ótimo, e a FME além de receber um vetor de movimento sub-ótimo, é simplificada através da técnica *Reduced FME*, originando um vetor de movimento possivelmente pior do que seria gerado se apenas uma das técnicas estivessem ativas.

Por outro lado, os casos onde o BD-BR da técnica *InterTech* é inferior à soma do BD-BR das técnicas constituintes está relacionado à hierarquia presente entre as técnicas constituintes. Se a técnica *Early Skip* realizar uma decisão prematura pelo modo *skip* em um determinado bloco, então nem a Busca Raster e nem a FME são realizadas para este bloco, de modo que as técnicas *Reduced SR* e *Reduced FME* não interferem na eficiência de codificação.

Por fim, uma análise sobre a coluna RCR mostra que no melhor caso a técnica *InterTech* obteve uma redução de complexidade relativa superior a 1200, na sequência de teste *Trolley*. Nesta sequência, foi possível atingir uma redução de complexidade igual a 26,78% com um valor de BD-BR de apenas 0,02%, indicando que a técnica *InterTech* foi capaz de simplificar significativamente o processo da codificação causando penalidades desprezíveis na eficiência de codificação.

Na Seção 4.3 foram discutidos alguns trabalhos que propõem otimizações para a predição interquadros de vídeos omnidirecionais, e dentre eles os trabalhos (RAY; JUNG; LARABI, 2018) e (LIU; SONG; ZHANG, 2019) propõem técnicas de redução de complexidade para a codificação de vídeos omnidirecionais na projeção ERP. Embora o trabalho (RAY; JUNG; LARABI, 2018) não utilize exatamente o mesmo conjunto de sequências de teste empregadas neste trabalho, e o trabalho (LIU; SONG; ZHANG, 2019) não relate as sequências de teste utilizadas, os resultados para ambos os trabalhos são apresentados na Tabela 13 em termos de redução de complexidade (RC), eficiência de codificação (BD-BR) e redução de complexidade relativa (RCR).

Como o trabalho (RAY; JUNG; LARABI, 2018) apresenta apenas o resultado de redução de complexidade médio para todas as sequências avaliadas, os resultados individuais de redução de complexidade não são apresentado na Tabela 13. Já o

Tabela 13 – Redução de complexidade e variação na eficiência de codificação dos trabalhos (RAY; JUNG; LARABI, 2018) e (LIU; SONG; ZHANG, 2019)

Res	Vídeo	InterTech			(RAY et al., 2018)			(LIU et al., 2019)		
		RC	BD-BR	RCR	RC	BD-BR	RCR	RC	BD-BR	RCR
4k	<i>AerialCity</i>	20,81%	0,79%	26,8	-	0%	-	-	-	-
	<i>PoleVault</i>	19,71%	0,31%	63,71	-	0%	-	-	-	-
	<i>DrivingInCity</i>	-	-	-	-	-0,1%	-	-	-	-
	<i>DrivingInCountry</i>	-	-	-	-	-0,4%	-	-	-	-
6k	<i>Balboa</i>	28,39%	0,85%	33,53	-	-	-	-	-	-
	<i>BranCastle</i>	14,54%	0,89%	16,28	-	-	-	-	-	-
	<i>Broadway</i>	27,63%	1,75%	15,76	-	-	-	-	-	-
	<i>Landing</i>	22,01%	1,13%	19,51	-	-	-	-	-	-
8k	<i>ChairliftRide</i>	17,82%	1,09%	16,42	-	-0,6%	-	-	-	-
	<i>GasLamp</i>	26,78%	0,05%	590,33	-	0%	-	-	-	-
	<i>Harbor</i>	28,57%	0,14%	201,43	-	0%	-	-	-	-
	<i>KiteFlite</i>	17,96%	0,30%	59,68	-	0,1%	-	-	-	-
8k	<i>SkateboardInLot</i>	23,06%	0,51%	45,34	-	0%	-	-	-	-
	<i>Trolley</i>	26,78%	0,02%	1231,12	-	0%	-	-	-	-
	<i>Glacier</i>	-	-	-	-	-0,9%	-	-	-	-
	<i>Bicycle</i>	-	-	-	-	-0,2%	-	-	-	-
<b>Média geral</b>		<b>22,84%</b>	<b>0,65%</b>	<b>193,29</b>	<b>15%</b>	<b>-0,2%</b>	<b>-75,0</b>	<b>32%</b>	<b>1,3%</b>	<b>24,6</b>

trabalho (LIU; SONG; ZHANG, 2019) apresenta apenas resultados médios, tanto para redução de complexidade quanto BD-BR.

Ao analisar a Tabela 13 percebe-se que a técnica proposta em (RAY; JUNG; LARABI, 2018) é capaz de atingir uma redução de complexidade de 15% com um BD-BR médio de -0,20%, o que leva a uma RCR igual a -75,0. Embora este trabalho consiga atingir resultados de redução de complexidade sem reduzir a eficiência de codificação, as técnicas propostas foram avaliados sobre o codificador *Joint Exploration Model*, que é um codificador de caráter exploratório utilizado para avaliar novos métodos de codificação antes do padrão sucessor ao HEVC começar seu desenvolvimento. Sendo assim, é possível que os resultados atingidos em (RAY; JUNG; LARABI, 2018) não estejam de acordo com o que aconteceria num padrão de codificação definido. Já o trabalho (LIU; SONG; ZHANG, 2019) é capaz de atingir uma redução de complexidade média de 32% com um BD-BR de 1,30%. Este trabalho apresenta resultados de redução de complexidade superiores à técnica *InterTech*, que obteve 22,84% de redução de complexidade, entretanto o valor de BD-BR em (LIU; SONG; ZHANG, 2019) é significativamente maior, visto que a técnica *InterTech* atingiu um BD-BR igual a 0,65%. Logo, a técnica *InterTech* é mais adequada em cenários onde é necessário atingir redução de complexidade sem impactar a eficiência de codificação de forma significativa.

Sendo assim, fica claro que a técnica *InterTech* é capaz de cumprir seu objetivo: reduzir a complexidade computacional da codificação de vídeos omnidirecionais sem penalizar de forma significativa a eficiência de codificação. Comparações com trabalhos relacionados mostram que a técnica *InterTech* é competitiva com outros trabalhos

da literatura, e dessa forma, é uma alternativa para cenários onde é necessário acelerar a codificação de vídeos omnidirecionais.

## 7.5 Resultados de *Reduced Intra Sizes*

Esta seção apresenta os resultados experimentais obtidos pela técnica de redução de complexidade *Reduced Intra Sizes*. Como a técnica *Reduced Intra Sizes* atua nas decisões de tamanho de bloco da predição intraquadro, a configuração de codificação *All Intra* é empregada nestas avaliações. Além disso, as demais técnicas de redução de complexidade foram desativadas visto que elas atuam somente na predição interquadros.

Como descrito na Seção 6.4, a técnica *Reduced Intra Sizes* é controlada por três parâmetros: *RefreshRate*, *minContrib* e *th*. O parâmetro *RefreshRate* controla o tamanho do *Frame Set*, ou seja, com que frequência as referências de custo RD serão atualizadas. O parâmetro *minContrib* determina quais tamanhos de bloco são elegíveis a passar pela redução de complexidade de acordo com a linha de CTU. Por fim, o parâmetro *th* é utilizado para admitir uma tolerância durante a definição de quais blocos terão seu sub-particionamento interrompido.

Diversas combinações destes parâmetros foram avaliadas com o objetivo de encontrar o conjunto que resulte na melhor relação entre redução de complexidade e eficiência de codificação. Por fim, o conjunto de parâmetros que obteve os melhores resultados foi utilizar *RefreshRate* igual a 4, *minContrib* igual a 0,35 e *th* igual a 1,01. Com base nestes parâmetros, a técnica atua em 75% dos quadros do vídeo (*RefreshRate* igual a quatro implica um quadro de referência e três quadros de atuação). Além disso, nos quadros onde a técnica *Reduced Intra Sizes* interfere na codificação, essa interferência ocorre apenas nas CUs cuja contribuição mínima supera 35% nas respectivas linhas de CTU. O Apêndice B apresenta a contribuição de cada tamanho de CU por linha de CTU.

Os resultados da técnica *Reduced Intra Sizes* com os parâmetros anteriormente definidos são apresentados na Tabela 14, onde a coluna RC representa a redução de complexidade, a coluna BD-BR representa a variação da eficiência de codificação enquanto que a coluna RCR representa a redução de complexidade relativa.

Ao analisar os dados de redução de complexidade da Tabela 14, é possível perceber que a técnica *Reduced Intra Sizes* alcançou até 12,16% de redução de complexidade, durante a codificação da sequência *AerialCity*. Enquanto isso, o pior desempenho no aspecto redução de complexidade se deu durante a codificação da sequência *Broadway*, onde a técnica *Reduced Intra Sizes* obteve 9,40% de redução de complexidade. Em média, a técnica *Reduced Intra Sizes* atingiu resultados de redução de complexidade iguais a 11,79%, 9,88% e 10,29% durante a codificação de sequências

Tabela 14 – Redução de complexidade e variação de eficiência de codificação da técnica *Reduced Intra Sizes*

Resolução	Vídeo	RC	BD-BR	RCR
4k	<i>AerialCity</i>	12,16%	0,37%	32,91
	<i>PoleVault</i>	11,43%	0,21%	54,11
6k	<i>Balboa</i>	9,75%	0,32%	30,26
	<i>BranCastle</i>	9,72%	0,16%	59,10
	<i>Broadway</i>	9,40%	0,55%	17,22
8k	<i>Landing</i>	10,64%	0,43%	24,54
	<i>ChairliftRide</i>	11,31%	0,68%	16,60
	<i>GasLamp</i>	10,18%	0,65%	15,60
	<i>Harbor</i>	9,70%	0,47%	20,48
	<i>KiteFlite</i>	10,72%	0,48%	22,25
	<i>SkateboardInlot</i>	10,12%	0,04%	270,18
	<i>Trolley</i>	9,72%	0,08%	117,41
	<b>Média 4k</b>	<b>11,79%</b>	<b>0,29%</b>	<b>43,51</b>
	<b>Média 6k</b>	<b>9,88%</b>	<b>0,37%</b>	<b>32,78</b>
	<b>Média 8k</b>	<b>10,29%</b>	<b>0,40%</b>	<b>77,09</b>
	<b>Média geral</b>	<b>10,41%</b>	<b>0,37%</b>	<b>56,72</b>

4k, 6k e 8k, respectivamente, enquanto que a média de redução de complexidade para todas as sequências foi igual a 10,31%.

Estes resultados também demonstram que a técnica *Reduced Intra Sizes* apresenta resultados de redução de complexidade bastante regulares, visto que todos eles estão entre 9,70% e 12,16%. Isso ocorre porque o parâmetro *minContrib* faz com que o mesmo subconjunto de CUs seja elegível para a redução de complexidade em todos os quadros do vídeo, e o custo RD durante a codificação de regiões co-localizadas costuma ser regular.

Já uma análise sobre a coluna BD-BR mostra que no melhor caso a técnica *Reduced Intra Sizes* acarretou um BD-BR igual a 0,04% na sequência *SkateboardInLot*, o que representa uma variação desprezível na eficiência de codificação. O pior caso no aspecto BD-BR ocorreu durante a codificação da sequência *ChairliftRide*, onde um BD-BR igual a 0,68% foi observado – uma variação pequena na eficiência de codificação. Em média, a técnica *Reduced Intra Sizes* acarretou valores de BD-BR iguais a 0,29%, 0,37% e 0,40% para sequências de resolução 4k, 6k e 8k, respectivamente, enquanto que o BD-BR médio para todas as sequências é igual a 0,37%.

As variações nos valores de BD-BR estão relacionadas ao quanto a técnica *Reduced Intra Sizes* interferiu nas decisões do codificador quando comparado às decisões tomadas pelo codificador sem modificações. Quando o tamanho de bloco ótimo para uma região é grande, e a técnica *Reduced Intra Sizes* descarta a avaliação de blocos menores nesta região, então ela está pouparando o codificador de um processamento

que não será aproveitado e a eficiência de codificação não é penalizada. Entretanto, caso uma região precise ser particionada em blocos pequenos para atingir a eficiência de codificação ótima e a técnica *Reduced Intra Sizes* impossibilita a avaliação destes blocos pequenos, então o codificador toma decisões sub-ótimas e a eficiência de codificação é penalizada.

Por fim, uma análise sobre a coluna RCR mostra que no melhor caso, a técnica *Reduced Intra Sizes* apresentou uma redução de complexidade relativa superior a 270, no vídeo *SkateboardInLot*, enquanto que o pior caso ocorreu durante a codificação do vídeo *GasLamp*, onde uma redução de complexidade relativa igual a 15,60 é observada. Em média, a técnica *Reduced Intra Sizes* apresentou valores de redução de complexidade relativa iguais a 43,51, 33,78 e 77,09 para vídeos de resolução 4k, 6k e 8k, respectivamente, enquanto que a redução de complexidade média para todas as sequências foi igual a 56,72.

Com exceção das sequências *SkateboardInLot* e *Trolley*, as demais sequências apresentaram reduções de complexidade relativa muito próximas. Isso se deve ao comportamento recorrente da redução de complexidade e do BD-BR, que variam pouco entre as múltiplas sequências codificadas. Este comportamento mostra que a técnica *Reduced Intra Sizes* tem um comportamento previsível durante a codificação da maioria das sequências omnidirecionais.

A partir dos resultados apresentados nesta seção, fica claro que a técnica *Reduced Intra Sizes* é capaz de atingir uma boa redução de complexidade durante a codificação de vídeos omnidirecionais, e além disso, a redução de complexidade é semelhante para todos os vídeos avaliados. Além disso, a técnica *Reduced Intra Sizes* causou pequenas e recorrentes penalizações na eficiência de codificação. Desta forma, a técnica *Reduced Intra Sizes* tem um comportamento similar para a maioria dos vídeos, e é adequada em sistemas de codificação que demandem uma redução de complexidade onde o resultado seja minimamente previsível.

## 7.6 Resultados das técnicas *IntraTech* integradas

Esta seção apresenta os resultados obtidos com as técnicas *IntraTech* integradas, isto é, tanto *Reduced Intra Modes* (STORCH et al., 2019) quanto *Reduced Intra Sizes*. A técnica *Reduced Intra Sizes* utilizou os mesmos parâmetros utilizados na Seção 7.5, isto é, *RefreshRate* igual a 4, *minContrib* igual a 0,35 e *th* igual a 1,01. A técnica *Reduced Intra Modes* é descrita em detalhes no trabalho (STORCH et al., 2019), mas de modo geral ela funciona como descrito a seguir.

A técnica *Reduced Intra Modes* utiliza uma abordagem em cinco bandas, assim como apresentado na Figura 19. A partir disso, quando uma PU nas bandas polares ou meio-polares está sendo codificada, a técnica *Reduced Intra Modes* reduz o

Tabela 15 – Redução de complexidade e variação de eficiência de codificação da técnica *IntraTech*

Resolução	Vídeo	RC	BD-BR	RCR
<b>4k</b>	<i>AerialCity</i>	24,95%	1,33%	18,79
	<i>PoleVault</i>	24,69%	0,75%	32,94
<b>6k</b>	<i>Balboa</i>	21,81%	1,28%	17,05
	<i>BranCastle</i>	25,18%	0,72%	35,08
	<i>Broadway</i>	21,66%	1,61%	13,44
<b>8k</b>	<i>Landing</i>	22,66%	1,99%	11,38
	<i>ChairliftRide</i>	24,06%	1,45%	16,59
	<i>GasLamp</i>	22,46%	1,78%	12,62
<b>8k</b>	<i>Harbor</i>	21,12%	1,47%	14,37
	<i>KiteFlite</i>	24,06%	1,07%	22,49
	<i>SkateboardInlot</i>	21,26%	0,58%	36,63
	<i>Trolley</i>	23,20%	0,53%	43,62
<b>Média 4k</b>		<b>24,82%</b>	<b>1,04%</b>	<b>25,86</b>
<b>Média 6k</b>		<b>22,83%</b>	<b>1,40%</b>	<b>19,24</b>
<b>Média 8k</b>		<b>22,69%</b>	<b>1,15%</b>	<b>24,39</b>
<b>Média geral</b>		<b>23,09%</b>	<b>1,21%</b>	<b>22,92</b>

número de modos de predição intraquadro que serão enviados a etapa de RDO para reduzir a complexidade da predição intraquadro. Para determinar quais modos devem ser enviados ao RDO, a técnica *Reduced Intra Modes* avalia uma distribuição estatística dos modos de predição por tamanho de bloco e banda, além das decisões tomadas pela etapa de RMD e MPM. Além disso, nas bandas polares a redução de complexidade é mais agressiva do que nas bandas meio-polares.

Os resultados da técnica *IntraTech* são apresentados na Tabela 15, onde a coluna RC representa a redução de complexidade atingida, a coluna BD-BR representa a variação na eficiência de codificação enquanto que a coluna RCR representa a redução de complexidade relativa.

Ao analisar os resultados de redução de complexidade da Tabela 15 percebe-se que a técnica *IntraTech* é capaz de atingir 25,18% de redução de complexidade no melhor caso, quando a sequência *BranCastle* é codificada. Por outro lado, o pior caso ocorreu durante a codificação da sequência *Harbor*, onde uma redução de complexidade igual a 21,12% é observada. Em média, a técnica *IntraTech* é capaz de atingir reduções de complexidade de 24,82%, 22,83% e 22,69% para sequências de resolução 4k, 6k e 8k, respectivamente, enquanto que a redução de complexidade média para todas as sequências é igual a 23,09%.

Quando os resultados de redução de complexidade da técnica *IntraTech* são comparados com a soma das reduções de complexidade das técnicas *Reduced Intra Sizes* e *Reduced Intra Modes*, percebe-se que a técnica *IntraTech* sempre obteve uma re-

dução de complexidade menor do que a soma da redução de complexidade de suas técnicas constituintes. Esse comportamento está associado à hierarquia existente entre as duas técnicas: sempre que a técnica *Reduced Intra Sizes* opta por interromper a avaliação de blocos menores, a técnica *Reduced Intra Modes* passa a atuar sobre um conjunto menor de blocos, e dessa forma, não é capaz de atingir todo o seu potencial de redução de complexidade.

Já uma análise da coluna BD-BR mostra que a técnica *IntraTech* acarretou perdas de eficiência de codificação em diversas magnitudes. O melhor caso ocorreu durante a codificação da sequência *Trolley*, onde um BD-BR igual a 0,53% foi observado, enquanto que o pior caso ocorreu durante a codificação da sequência *Landing*, onde o BD-BR foi igual a 1,99%. Em média, a técnica *IntraTech* acarretou um BD-BR igual a 1,04%, 1,40% e 1,15% para sequências de resolução 4k, 6k e 8k, enquanto que o BD-BR médio para todos os vídeos foi igual a 1,21%.

De forma similar ao que ocorreu nos resultados de redução de complexidade, o BD-BR da técnica *IntraTech* é superior ao BD-BR de suas técnicas constituintes em todos os vídeos avaliados, e este comportamento pode estar associado a dois comportamentos. O primeiro deles é que quando a técnica *Reduced Intra Modes* descarta a avaliação de alguns modos de predição intraquadro, isso altera o custo RD da CU sendo codificada e esta alteração pode fazer com que a técnica *Reduced Intra Sizes* interrompa o particionamento de CUs com maior frequência. O segundo comportamento está associado ao fato que quando apenas uma das técnicas é utilizada, é possível obter-se um modo de predição ótimo num particionamento sub-ótimo (técnica *Reduced Intra Sizes*) ou um modo de predição sub-ótimo para um particionamento ótimo (técnica *Reduced Intra Modes*), enquanto que a utilização de ambas as técnicas pode acarretar em um modo de predição sub-ótimo num particionamento sub-ótimo, o que é mais prejudicial para a eficiência de codificação.

Por fim, uma análise sobre a coluna RCR mostra que a técnica *IntraTech* apresenta um comportamento regular no aspecto redução de complexidade relativa. No melhor caso, a técnica *IntraTech* obteve uma redução de complexidade relativa igual a 43,62 na sequência *Trolley*, enquanto que o pior caso ocorreu na sequência *Landing*, onde uma redução de complexidade relativa igual a 11,38 foi observada.

Na Seção 4.2 foram discutidos alguns trabalhos que propõem otimizações para a predição intraquadro de vídeos omnidirecionais, e dentre eles os trabalhos (WANG et al., 2017) e (ZHANG et al., 2019) propõem técnicas de redução de complexidade para a codificação de vídeos omnidirecionais na projeção ERP. Sendo assim, a Tabela 16 apresenta os resultados de redução de complexidade e eficiência de codificação dos trabalhos (WANG et al., 2017) e (ZHANG et al., 2019). Na Tabela 16 algumas células estão vazias porque a técnica *IntraTech* e os trabalhos propostos em (WANG et al., 2017) e (ZHANG et al., 2019) não utilizam exatamente o mesmo conjunto de

Tabela 16 – Redução de complexidade e variação na eficiência de codificação dos trabalhos (WANG et al., 2017) e (ZHANG et al., 2019)

Res	Vídeo	<i>IntraTech</i>			(ZHANG et al., 2019)			(WANG et al., 2017)		
		RC	BD	RCR	RC	BD-BR	RCR	Time	BD-BR	Tradeoff
4k	<i>AerialCity</i>	24,95%	1,33%	18,79	54,0%	1,2%	45,0	26,9%	0,38%	70,8
	<i>PoleVault</i>	24,69%	0,75%	32,94	51,0%	1,2%	42,5	24,4%	0,37%	66,0
	<i>DrivingInCity</i>	-	-	-	59,0%	1,5%	39,3	30,1%	0,15%	200,5
	<i>DrivingInCountry</i>	-	-	-	53,0%	0,8%	66,3	25,9%	0,20%	129,5
6k	<i>Balboa</i>	21,81%	1,28%	17,05	57,0%	1,3%	43,9	-	-	-
	<i>BranCastle</i>	25,18%	0,72%	35,08	46,0%	1,1%	41,8	-	-	-
	<i>Broadway</i>	21,66%	1,61%	13,44	53,0%	1,8%	29,4	-	-	-
	<i>Landing</i>	22,66%	1,99%	11,38	52,0%	1,7%	30,6	-	-	-
8k	<i>ChairliftRide</i>	24,06%	1,45%	16,59	47,0%	1,0%	47,0	25,1%	0,17%	147,4
	<i>GasLamp</i>	22,46%	1,78%	12,62	51,0%	2,6%	19,6	21,5%	0,23%	93,5
	<i>Harbor</i>	21,12%	1,47%	14,37	52,0%	2,1%	24,8	23,9%	0,14%	170,6
	<i>KiteElite</i>	24,06%	1,07%	22,49	44,0%	1,2%	36,7	20,4%	0,38%	53,7
	<i>SkateboardInlot</i>	21,26%	0,58%	36,63	54,0%	1,4%	38,6	23,8%	0,09%	264,7
	<i>Trolley</i>	23,30%	0,53%	43,62	46,0%	1,0%	46,0	23,0%	0,27%	85,2
	<i>SkateboardTrick</i>	-	-	-	59,0%	1,1%	53,6	-	-	-
	<i>Train</i>	-	-	-	56,0%	0,5%	112,0	-	-	-
<b>Média</b>		<b>23,09%</b>	<b>1,21%</b>	<b>22,92</b>	<b>53,00%</b>	<b>1,3%</b>	<b>44,8</b>	<b>24,5%</b>	<b>0,24%</b>	<b>128,2</b>

sequências de teste.

Quando os resultados da Tabela 16 são analisados, percebe-se que a técnica de (ZHANG et al., 2019) atinge uma redução de complexidade média de 53% com um BD-BR médio de 1,3%, o que representa uma RCR de 44,81. Já o trabalho (WANG et al., 2017) atinge uma redução de complexidade média de 24,50% junto de um BD-BR médio de 0,24%, o que significa uma RCR média de 128,2. Quando estes resultados são comparados aos resultados obtidos pela técnica *IntraTech*, que resultou numa redução de complexidade média de 23,09% com um BD-BR de 1,21%, percebe-se que ambos os trabalhos relacionados obtiveram uma performance melhor do que a técnica desenvolvida. O trabalho (ZHANG et al., 2019) atingiu uma redução de complexidade significativamente maior do que a técnica *IntraTech* com um pequeno acréscimo de BD-BR, enquanto que o trabalho (WANG et al., 2017) atingiu uma redução de complexidade próxima da técnica *IntraTech*, mas com um BD-BR menor.

Embora estes trabalhos sejam capazes de atingir resultados superiores à técnica *IntraTech* desenvolvida, os trabalhos (ZHANG et al., 2019) e (WANG et al., 2017) são compostos por técnicas majoritariamente genéricas de redução de complexidade que são aplicadas a todas as regiões de vídeos omnidirecionais na mesma intensidade, e que possivelmente alcançariam resultados semelhantes caso aplicadas a vídeos convencionais, enquanto que a técnica *IntraTech* é adaptativa às regiões do vídeo e atua com intensidade proporcional às distorções causadas pela projeção ERP. Sendo assim, a técnica *IntraTech* pode ser aliada às técnicas genéricas de redução de complexidade para obter um resultado final mais eficiente, além de ter o seu próprio desempenho melhorado caso uma implementação com mais bandas – e que modela a distorção de forma mais adequada – fosse utilizada.

Sendo assim, a técnica *IntraTech* se mostrou uma boa candidata para sistemas que demandem uma aceleração no processo de codificação, visto que é capaz de atingir reduções de complexidade significativas com um impacto aceitável na eficiência de codificação. Embora ela não seja capaz de atingir uma performance superior a trabalhos semelhantes que utilizam técnicas genéricas de redução de complexidade, ela explora as características específicas de vídeos omnidirecionais na projeção ERP e pode ser aliada a maioria das técnicas de redução de complexidade genéricas para obter um melhor desempenho.

## 8 CONCLUSÃO

Este trabalho apresentou uma série de técnicas de redução de complexidade para a codificação de vídeos omnidirecionais através da exploração das distorções de textura causadas pela projeção ERP, que é utilizada para permitir que vídeos omnidirecionais, de natureza esférica, possam ser codificados por padrões de codificação de vídeos convencionais. A principal distorção causada pela projeção ERP é o esticamento horizontal das texturas nas regiões superior e inferior do quadro, o que causa um comportamento característico nas ferramentas de codificação de vídeo.

Dentre os comportamentos característicos causados pela projeção ERP, um deles se manifesta nas escolhas do modo de codificação *skip*. O modo *skip* costuma ser empregado em regiões bastante homogêneas do vídeo, e o esticamento intenso causado pela projeção ERP faz com que regiões específicas do quadro tenham uma tendência maior a serem codificadas com o modo *skip* – algo que não ocorre na codificação de vídeos convencionais. Além disso, análises estatísticas também mostraram que estas ocorrências estão diretamente relacionadas com a alta homogeneidade presente nos blocos destas regiões.

Já durante a estimativa de movimento, foi constatado que os vetores de movimento resultantes da Busca Raster – etapa mais custosa do algoritmo de casamento de blocos – apresentam características distintas de acordo com a região do vídeo omnidirecional sendo codificado, enquanto que durante a codificação de vídeos convencionais todas as regiões são codificadas com vetores de movimento semelhantes. Além disso, outra análise apontou que devido ao esticamento horizontal causado pela projeção ERP, blocos horizontalmente vizinhos dentro da área de busca são redundantes, e não é necessário avaliá-los todos para obter vetores de movimento ótimos.

Além disso, o esticamento horizontal causado pela projeção ERP é realizado através da interpolação de amostras, isto é, são geradas amostras em posições horizontalmente fracionárias na superfície esférica para mapeá-las para o plano retangular. Sendo assim, como amostras horizontalmente fracionárias já são criadas durante a projeção, a estimativa de movimento fracionária causa pequenos ganhos de eficiência de codificação quando aplicada às regiões distorcidas.

Por fim, as distorções causadas pela projeção ERP fazem com que as regiões próximas ao limite superior e inferior do quadro sejam naturalmente mais homogêneas, e esta homogeneidade é reduzida conforme aproxima-se do centro do quadro. Esta característica faz com que durante a predição intraquadro, as regiões superior e inferior do quadro apresentem uma tendência a serem codificados com blocos maiores, e conforme aproxima-se do centro do quadro, a utilização de blocos maiores passa a ser menor, enquanto que blocos menores se tornam mais frequentes.

A partir destas constatações, uma série de técnicas de redução de complexidade para a codificação de vídeos omnidirecionais foi desenvolvida. Uma destas técnicas é aplicada na predição interquadros e se chama *InterTech*. A técnica *InterTech* é composta por três técnicas menores: *Early Skip*, *Reduced SR* e *Reduced FME*. A outra técnica é aplicada na predição intraquadro e se chama *IntraTech*. A técnica *IntraTech* é composta por duas técnicas menores: *Reduced Intra Sizes* e *Reduced Intra Modes*, esta última sendo herdada de um trabalho prévio (STORCH et al., 2019).

A técnica *Early Skip* realiza uma decisão prematura pelo modo *skip* com base na homogeneidade do bloco sendo codificado e a posição vertical deste dentro do quadro. Resultados experimentais demonstraram que é possível atingir elevadas reduções de complexidade com um impacto insignificante na eficiência de codificação.

Já a técnica *Reduced SR* consiste em reduzir a largura da área de busca durante a Busca Raster de acordo com o esticamento causado pela projeção ERP, isto é, onde há maior esticamento, menor é a área de busca. Quando esta técnica é utilizada, é possível atingir uma pequena redução de complexidade na codificação de vídeos omnidirecionais, entretanto, a eficiência de codificação não é alterada.

A técnica *Reduced FME* consiste em alterar a precisão horizontal da FME de acordo com o esticamento causado pela projeção ERP, de modo que onde há maior distorção, menor é a precisão. Avaliações experimentais da técnica *Reduced FME* apontaram que está é capaz de atingir reduções de complexidade significativas ao custo de uma pequena redução na eficiência de codificação.

A técnica *Reduced Intra Sizes* descarta a avaliação de alguns tamanhos de bloco durante a predição intraquadro de acordo com a intensidade do esticamento causado pela projeção ERP e a correlação temporal entre quadros vizinhos do vídeo. Através disso, a técnica *Reduced Intra Sizes* é capaz de atingir reduções significativas na complexidade de codificação de vídeos omnidirecionais, com pequenas perdas de eficiência de codificação.

A partir do desempenho obtido pelas técnicas desenvolvidas, é possível concluir que explorar as distorções causadas pela projeção ERP – e seu respectivo impacto nas ferramentas de codificação – é uma boa alternativa para reduzir a complexidade de diversas ferramentas de codificação de vídeo, visto que é possível atingir elevadas reduções de complexidade com impactos mínimos na eficiência de codificação.

Além disso, vídeos omnidirecionais têm ganhado muita popularidade ultimamente com o surgimento de diversas aplicações (desde entretenimento até negócios imobiliários) e a popularização de dispositivos capazes de suportar realidade virtual e vídeos imersivos. Sendo assim, fica claro que existe uma demanda por soluções para a codificação de vídeos omnidirecionais para facilitar a distribuição deste tipo de conteúdo e suprir com eficiência as demandas atuais e que devem surgir no futuro.

## 9 TRABALHOS SUBMETIDOS E PUBLICADOS

### 9.1 Trabalhos na ÁREA da Dissertação

#### 9.1.1 The Impacts of Equirectangular 360-degrees Videos in the Intra-Frame Prediction of HEVC

Iago Storch, Luis A. da Silva Cruz, Luciano Agostini, Bruno Zatt, Daniel Palomino  
Journal of Integrated Circuits and Systems (**JICS**), 2019  
Qualis B2

#### 9.1.2 Spatially Adaptive Intra Mode pre-Selection for ERP 360 Video Coding (aceito para publicação)

Iago Storch, Bruno Zatt, Luciano Agostini, Guilherme Correa, Luis. A. da Silva Cruz, Daniel Palomino  
International Conference on Acoustics, Speech, and Signal Processing (**ICASSP**), 2020  
Qualis A1

### 9.2 Trabalhos Fora da ÁREA da Dissertação

#### 9.2.1 Speedup evaluation of HEVC parallel video coding using Tiles

Iago Storch, Daniel Palomino, Bruno Zatt, Luciano Agostini  
Journal of Real-Time Image Processing (**JRTIP**), 2019  
Qualis B1

#### 9.2.2 Memory-aware Workload Balancing Technique based on Decision Trees for Parallel HEVC Video Coding (*submetido*)

Iago Storch, Bruno Zatt, Luciano Agostini, Guilherme Correa, Daniel Palomino  
Journal of Integrated Circuits and Systems (**JICS**), 2020  
Qualis B2

### 9.2.3 HEVC Video Coding Using Decision Trees for a Memory-Friendly Tiles Workload Balance (*Best Paper Award*)

Iago Storch, Bruno Zatt, Luciano Agostini, Guilherme Correa, Daniel Palomino

Simpósio Sul de Microeletrônica (**SIM**), 2019

Sem Qualis

## REFERÊNCIAS

- BJØNTEGAARD, G. VCEG-M33: Calculation of Average PSNR Differences between RD curves. **Video Coding Experts Group (VCEG)**, [S.I.], 2001.
- BOYCE, J. **JVET-L0647**: BoG report on 360 video. [S.I.: s.n.], 2018. Document.
- BOYCE, J.; ALSHINA, E.; ABBAS, A.; YE, Y. JVET common test conditions and evaluation procedures for 360 video. **JVET-H1030, Chengdu, CN**, [S.I.], 2017.
- CISCO. **Cisco Visual Networking Index**: Forecast and Trends, 2017–2022 White Paper. Disponível em: <<https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html>>. Acesso em: 2019-06-16.
- CISCO. **Cisco Visual Networking Index**: Global Mobile Data Traffic Forecast Update, 2017–2022 White Paper. Disponível em: <<https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-738429.html>>. Acesso em: 2019-06-16.
- CONCEIÇÃO, R. Ávila. **Early Skip/DIS**: uma heurística para redução de complexidade no codificador de mapas de profundidade do 3D-HEVC. 2016. Dissertação de Mestrado (Ciência da Computação) — Universidade Federal de Pelotas, Pelotas.
- CORRÊA, D. S. **Exploração no Projeto de Hierarquia de Memória para Codificação de Vídeo Energeticamente Eficiente no Padrão HEVC**. 2018. Trabalho de Conclusão de Curso (Engenharia de Computação) — Universidade Federal de Pelotas, Pelotas.
- CORREA, G.; ASSUNCAO, P. A.; AGOSTINI, L. V.; SILVA CRUZ, L. A. da. Fast HEVC Encoding Decisions Using Data Mining. **IEEE Transactions on Circuits and Systems for Video Technology**, [S.I.], v.25, n.4, p.660–673, April 2015.
- DESIMONE, F.; FROSSARD, P.; BIRKBECK, N.; ADSUMILLI, B. Deformable block-based motion estimation in omnidirectional image sequences. In: **IEEE 19TH INTER**

NATIONAL WORKSHOP ON MULTIMEDIA SIGNAL PROCESSING (MMSP), 2017., 2017. **Anais...** [S.I.: s.n.], 2017. p.1–6.

DETREMPE, K. **Hospital-wide access to virtual reality alleviates pain and anxiety for pediatric patients.** Disponível em: <<https://www.stanfordchildrens.org/en/about/news/releases/2017/virtual-reality-alleviates-pain-anxiety>>. Acesso em: 2019-06-16.

FRANKLIN, P. **Nintendo shows off 3D portable game device.** Disponível em: <<https://www.reuters.com/article/us-e3-nintendo/nintendo-shows-off-3d-portable-game-device-idUSTRE65E4LP20100615>>. Acesso em: 2019-06-16.

GHANBARI, M. **Standard codecs:** Image compression to advanced video coding. [S.I.]: let, 2003. n.49.

GLEB, B. **How Virtual Reality is Transforming the Real State Industry.** Disponível em: <<https://rubygarage.org/blog/virtual-reality-in-real-estate>>. Acesso em: 2019-06-16.

GONÇALVES, P. et al. Multiple early-termination scheme for TZ search algorithm based on data mining and decision trees. In: IEEE 19TH INTERNATIONAL WORKSHOP ON MULTIMEDIA SIGNAL PROCESSING (MMSP), 2017., 2017. **Anais...** [S.I.: s.n.], 2017. p.1–6.

GONÇALVES, P. H. R. **Esquema Híbrido para Redução do Esforço Computacional da Estimação de Movimento no Codificador de Vídeo HEVC.** 2018. Trabalho de Conclusão de Curso (Ciência da Computação) — Universidade Federal de Pelotas, Pelotas.

GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing (3rd Edition).** Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006.

GOPRO. **GoPro Omni.** Disponível em: <<https://pt.gopro.com/help/articles/block/What-is-Omni>>. Acesso em: 2019-06-18.

GOPRO. **GoPro HERO4 Black.** Disponível em: <<https://pt.gopro.com/help/HERO4-Black>>. Acesso em: 2019-06-18.

HANHART, P.; LIN, J. L. **JVET-K0033:** CE13: Summary report on projection formats. [S.I.: s.n.], 2018. Document.

HANHART, P.; LIN, J. L.; PUJARA, C. **JVET-L0033:** CE13: Summary report on coding tools for 360 omnidirectional video. [S.I.: s.n.], 2018. Document.

HANHART, P.; LIN, J. L.; PUJARA, C. **JVET-M0033**: CE13: Summary report on coding tools for 360 omnidirectional video. [S.I.: s.n.], 2019. Document.

HANNUKSELA, M. M.; WANG, Y.; HOURUNRANTA, A. An Overview of the OMAF Standard for 360 Video. In: DATA COMPRESSION CONFERENCE (DCC), 2019., 2019. **Anais...** [S.I.: s.n.], 2019. p.418–427.

HOOFT, J. v. der et al. Optimizing Adaptive Tile-Based Virtual Reality Video Streaming. In: IFIP/IEEE SYMPOSIUM ON INTEGRATED NETWORK AND SERVICE MANAGEMENT (IM), 2019., 2019. **Anais...** [S.I.: s.n.], 2019. p.381–387.

HUFFMAN, D. A. A Method for the Construction of Minimum-Redundancy Codes. **Proceedings of the IRE**, [S.I.], v.40, n.9, p.1098–1101, Sep. 1952.

JCT-VC. Recommendation ITU-T H. 265. **International Standard ISO/IEC**, [S.I.], p.23008–2, 2013.

JCT-VC. **HEVC Test Model version 16.16**. Disponível em: <[https://hevc.hhi.fraunhofer.de/svn/svn\\_HEVCSoftware/tags/HM-16.16/](https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.16/)>. Acesso em: 2018-11-19.

JVET. **Versatile Video Coding (Draft 5)**. [S.I.: s.n.], 2019.

JVT. Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification. **ITU-T Rec. H. 264**, [S.I.], 2003.

KHAN, M. U. K.; SHAFIQUE, M.; HENKEL, J. An adaptive complexity reduction scheme with fast prediction unit decision for HEVC intra encoding. In: IEEE INTERNATIONAL CONFERENCE ON IMAGE PROCESSING, 2013., 2013. **Anais...** [S.I.: s.n.], 2013. p.1578–1582.

LG. **LG 360 CAM**. Disponível em: <<https://www.lg.com/br/acessorios-para-celular/lgr105>>. Acesso em: 2019-06-18.

LIU, Z.; SONG, P.; ZHANG, M. A CU Split Early Termination Algorithm Based KNN for 360-Degree Video. **Data Compression Conference Proceedings**, [S.I.], v.2019-March, n.Dcc, p.594, 2019.

LUZ, G. F. G. **Omnidirectional Video**: Adaptive Coding based on Saliency. 2017. Master of Science Degree in Electric and Computer Engineering — Instituto Superior Técnico, Lisboa.

MPEG. **Coded representation of immersive media - Part 2**: omnidirectional media format. [S.I.: s.n.], 2019. Document.

OCULUS. **Oculus**. Disponível em: <<https://www.oculus.com/>>. Acesso em: 2019-06-17.

OHM, J. et al. Comparison of the Coding Efficiency of Video Coding Standards—Including High Efficiency Video Coding (HEVC). **IEEE Transactions on Circuits and Systems for Video Technology**, [S.I.], v.22, n.12, p.1669–1684, Dec 2012.

PAN, Z. et al. Fast Motion Estimation Based on Content Property for Low-Complexity H.265/HEVC Encoder. **IEEE Transactions on Broadcasting**, [S.I.], v.62, n.3, p.675–684, Sep. 2016.

PISOFTECH. **Pilot Era**. Disponível em: <<https://pilot.labpano.com/>>. Acesso em: 2019-06-18.

POLLACK, J. Displays of A Different Stripe. **IEEE Spectrum**, [S.I.], v.43, n.8, p.40–44, Aug 2006.

RAY, B.; JUNG, J.; LARABI, M. A Low-Complexity Video Encoder for Equirectangular Projected 360 Video Content. In: IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING (ICASSP), 2018., 2018. **Anais...** [S.I.: s.n.], 2018. p.1723–1727.

RICHARDSON, I. E. **Video Codec Design**: Developing Image and Video Compression Systems. New York, NY, USA: John Wiley & Sons, Inc., 2002.

ROZAKIS, R. **Using VR 360 Video as a Teaching Tool**. Disponível em: <<https://corp.kaltura.com/blog/using-vr-360-video-as-a-teaching-tool/>>. Acesso em: 2019-06-16.

SAMSUNG. **Samsung Gear VR**. Disponível em: <<https://www.samsung.com.br/wearables/vr/>>. Acesso em: 2019-06-17.

SAMSUNG. **Samsung Gear 360**. Disponível em: <<https://www.samsung.com/br/wearables/gear-360-2017/>>. Acesso em: 2019-06-18.

SHARMAN, K.; SUEHRING, K. Common test conditions. **JCTVC-AC1100, Macao, CN**, [S.I.], 2017.

SHI, Y. Q.; SUN, H. **Image and Video Compression for Multimedia Engineering**: Fundamentals, Algorithms, and Standards. 2nd.ed. Boca Raton, FL, USA: CRC Press, Inc., 2008.

STORCH, I. C. **Otimização das Ferramentas de Codificação de Vídeo do Padrão HEVC para Aplicações com Vídeos Omnidirecionais**. 2018. Trabalho de Conclusão de Curso (Ciência da Computação) — Universidade Federal de Pelotas, Pelotas.

STORCH, I. et al. The Impacts of Equirectangular 360-degrees Videos in the Intra-Frame Prediction of HEVC. **Journal of Integrated Circuits and Systems**, [S.I.], v.14, n.1, p.1–10, 2019.

STORCH, I. et al. FastIntra360: A Fast Intra-Prediction Technique for 360-Degrees Video Coding. In: DATA COMPRESSION CONFERENCE (DCC), 2019., 2019. **Anais...** [S.I.: s.n.], 2019. p.605–605.

SULLIVAN, G. J. et al. Overview of the high efficiency video coding(HEVC) standard. **IEEE Transactions on circuits and systems for video technology**, zzzz, v.22, n.12, p.1649–1668, 2012.

SULLIVAN, G. J.; WIEGAND, T. Rate-distortion optimization for video compression. **IEEE signal processing magazine**, [S.I.], v.15, n.6, p.74–90, 1998.

SULLIVAN, G.; OHM, J.-R. **JVET-J1000**: Meeting Report of the 10th meeting of the Joint Video Experts Team (JVET), San Diego, US, 10–20 Apr. 2018. [S.I.: s.n.], 2018. Document.

SZE, V.; BUDAGAVI, M.; SULLIVAN, G. J. High efficiency video coding (HEVC). **Integrated Circuit and Systems, Algorithms and Architectures**. Springer, [S.I.], v.39, p.40, 2014.

SZELISKI, R. et al. Image alignment and stitching: A tutorial. **Foundations and Trends® in Computer Graphics and Vision**, [S.I.], v.2, n.1, p.1–104, 2007.

VISBIT. **Why do all the 360 VR videos today look so pixelated?** Disponível em: <<https://medium.com/visbit/why-do-all-the-360-vr-videos-today-look-so-pixelated-b1ab3cba6f95>>. Acesso em: 2019-06-16.

VISHWANATH, B.; NANJUNDASWAMY, T.; ROSE, K. Rotational motion model for temporal prediction in 360 video coding. In: IEEE 19TH INTERNATIONAL WORKSHOP ON MULTIMEDIA SIGNAL PROCESSING (MMSP), 2017., 2017. **Anais...** [S.I.: s.n.], 2017. p.1–6.

VIVE. **Vive**. Disponível em: <<https://www.vive.com/>>. Acesso em: 2019-06-17.

WANG, Y.; LI, Y.; YANG, D.; CHEN, Z. A fast intra prediction algorithm for 360-degree equirectangular panoramic video. In: IEEE VISUAL COMMUNICATIONS AND IMAGE PROCESSING, 2017., 2017. **Anais...** [S.I.: s.n.], 2017. p.1–4.

- WITTEN, I. H.; NEAL, R. M.; CLEARY, J. G. Arithmetic Coding for Data Compression. **Commun. ACM**, New York, NY, USA, v.30, n.6, p.520–540, June 1987.
- YE, Y.; ALSHINA, E.; BOYCE, J. Algorithm descriptions of projection format conversion and video quality metrics in 360Lib. **Joint Video Exploration Team of ITU-T SG**, [S.I.], v.16, 2017.
- YU, M.; LAKSHMAN, H.; GIROD, B. A framework to evaluate omnidirectional video coding schemes. In: IEEE INTERNATIONAL SYMPOSIUM ON MIXED AND AUGMENTED REALITY (ISMAR), 2015., 2015. **Anais...** [S.I.: s.n.], 2015. p.31–36.
- ZARE, A.; AMINLOU, A.; HANNUKSELA, M. M.; GABBOUJ, M. HEVC-compliant Tile-based Streaming of Panoramic Video for Virtual Reality Applications. In: ACM INTERNATIONAL CONFERENCE ON MULTIMEDIA, 24., 2016, New York, NY, USA. **Proceedings...** ACM, 2016. p.601–605. (MM '16).
- ZATT, B.; SHAFIQUE, M.; BAMPI, S.; HENKEL, J. An adaptive early skip mode decision scheme for multiview video coding. In: PICTURE CODING SYMPOSIUM, 28., 2010. **Anais...** [S.I.: s.n.], 2010. p.42–45.
- ZHANG, M. et al. Fast intra algorithm based on texture characteristics for 360 videos. **Eurasip Journal on Image and Video Processing**, [S.I.], v.2019, n.1, 2019.

## APÊNDICE A VARIÂNCIAS DE CORTE DE ACORDO COM RESOLUÇÃO, NÚMERO DE BANDAS, *CUTOFF*, TAMANHO DE BLOCO E QP

3328×1664, três bandas, bandas polares, QP22

<i>Cutoff</i>	Tamanho de CU			
	64×64	32×32	16×16	8×8
0,05	2	1	1	1
0,10	2	1	1	1
0,15	3	2	1	1
0,20	4	2	1	1
0,25	5	3	2	1
0,30	7	3	2	1
0,35	9	4	2	2
0,40	12	5	3	2
0,45	16	8	5	3
0,50	20	10	8	4
0,55	40	20	8	4
0,60	40	30	12	6
0,65	60	40	16	8
0,70	80	50	24	12
0,75	120	70	32	18
0,80	140	90	52	26
0,85	200	140	84	48
0,90	280	210	156	100

3328×1664, três bandas, bandas polares, QP27

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	3	2	1	1
0,15	6	2	1	1
0,20	9	3	2	1
0,25	12	4	2	1
0,30	17	6	3	2
0,35	23	9	4	2
0,40	34	13	5	3
0,45	47	19	7	4
0,50	80	30	12	4
0,55	100	40	16	6
0,60	120	60	24	8
0,65	160	80	32	12
0,70	200	110	44	18
0,75	240	150	72	28
0,80	320	210	116	46
0,85	400	300	192	88
0,90	540	450	328	182

3328×1664, três bandas, bandas polares, QP32

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	4	2	1	1
0,15	6	2	1	1
0,20	10	3	2	1
0,25	15	5	2	1
0,30	20	7	3	2
0,35	31	10	4	2
0,40	44	16	6	3
0,45	64	24	8	4
0,50	100	40	12	6
0,55	120	50	20	8
0,60	160	70	24	10
0,65	200	100	36	14
0,70	240	140	56	20
0,75	320	190	88	30
0,80	380	260	140	54
0,85	500	360	228	100
0,90	680	530	372	202

3328×1664, três bandas, bandas polares, QP37

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	4	2	1	1
0,15	7	2	1	1
0,20	11	4	2	1
0,25	16	5	2	1
0,30	23	7	3	2
0,35	36	11	4	2
0,40	53	18	6	3
0,45	75	26	9	4
0,50	100	40	12	6
0,55	140	60	20	8
0,60	180	80	28	10
0,65	220	110	40	14
0,70	300	160	60	20
0,75	360	210	96	32
0,80	440	290	152	56
0,85	560	400	240	106
0,90	780	580	388	388

3328×1664, três bandas, bandas polares, outro QP

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	3	2	1	1
0,15	5	2	1	1
0,20	9	3	2	1
0,25	12	4	2	1
0,30	17	6	3	1
0,35	24	9	4	2
0,40	37	13	5	3
0,45	53	20	7	4
0,50	80	30	12	6
0,55	100	40	16	6
0,60	140	60	24	8
0,65	180	80	32	12
0,70	200	110	44	18
0,75	280	160	68	26
0,80	360	230	116	40
0,85	460	320	192	84
0,90	640	480	332	176

4096×2048, três bandas, bandas polares, QP22

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	1	1	1	1
0,20	1	1	1	1
0,25	1	1	1	1
0,30	2	1	1	1
0,35	2	1	1	1
0,40	2	2	1	1
0,45	2	2	2	1
0,50	2	2	2	2
0,55	2	2	2	2
0,60	2	2	2	2
0,65	3	2	2	2
0,70	3	2	2	2
0,75	4	3	2	2
0,80	5	3	3	3
0,85	7	4	3	4
0,90	9	5	5	5

4096×2048, três bandas, bandas polares, QP27

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	1	1	1	1
0,20	2	1	1	1
0,25	2	1	1	1
0,30	2	2	1	1
0,35	2	2	2	1
0,40	2	2	2	1
0,45	2	2	2	2
0,50	3	2	2	2
0,55	3	2	2	2
0,60	4	3	2	2
0,65	4	3	3	3
0,70	5	4	4	3
0,75	6	5	5	4
0,80	7	6	6	5
0,85	9	7	7	7
0,90	11	9	10	9

4096×2048, três bandas, bandas polares, QP32

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	2	1	1	1
0,20	2	1	1	1
0,25	2	2	1	1
0,30	2	2	2	1
0,35	2	2	2	1
0,40	3	2	2	2
0,45	3	2	2	2
0,50	4	3	2	2
0,55	5	3	3	2
0,60	6	4	3	2
0,65	6	5	4	3
0,70	8	6	5	4
0,75	9	7	6	5
0,80	10	9	8	6
0,85	12	11	10	8
0,90	15	14	13	11

4096×2048, três bandas, bandas polares, QP37

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	2	1	1	1
0,20	2	2	1	1
0,25	2	2	1	1
0,30	2	2	2	1
0,35	3	2	2	1
0,40	4	2	2	2
0,45	5	3	2	2
0,50	5	3	2	2
0,55	6	4	3	2
0,60	8	5	4	2
0,65	9	6	5	3
0,70	10	7	6	4
0,75	12	9	7	5
0,80	14	11	9	6
0,85	16	13	11	8
0,90	20	17	15	12

4096×2048, três bandas, bandas polares, outro QP

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	2	1	1	1
0,20	2	1	1	1
0,25	2	2	1	1
0,30	2	2	2	1
0,35	2	2	2	1
0,40	3	2	2	1
0,45	3	2	2	2
0,50	4	2	2	2
0,55	4	3	2	2
0,60	5	3	3	2
0,65	6	4	3	3
0,70	7	5	4	3
0,75	9	6	5	4
0,80	10	8	7	5
0,85	12	10	9	7
0,90	16	13	12	10

3328×1664, cinco bandas, bandas polares, QP22

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	2	1	1	1
0,15	2	2	1	1
0,20	3	2	1	1
0,25	4	2	1	1
0,30	6	3	2	1
0,35	8	4	2	2
0,40	10	5	3	2
0,45	12	6	4	3
0,50	20	10	8	2
0,55	20	10	8	4
0,60	40	20	12	6
0,65	60	30	16	8
0,70	80	40	20	10
0,75	100	50	28	14
0,80	120	70	40	22
0,85	160	100	60	38
0,90	200	160	112	76

3328×1664, cinco bandas, bandas polares, QP27

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	3	2	1	1
0,15	5	2	1	1
0,20	8	3	1	1
0,25	11	4	2	1
0,30	14	5	3	1
0,35	20	7	4	2
0,40	27	11	5	3
0,45	41	16	7	3
0,50	60	30	12	4
0,55	80	40	16	6
0,60	100	50	20	8
0,65	140	70	28	12
0,70	180	90	40	16
0,75	220	130	60	24
0,80	300	190	100	40
0,85	380	260	164	74
0,90	500	380	268	146

3328×1664, cinco bandas, bandas polares, QP32

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	3	2	1	1
0,15	6	2	1	1
0,20	9	3	2	1
0,25	13	4	2	1
0,30	18	6	3	1
0,35	25	9	4	2
0,40	40	14	5	3
0,45	57	21	7	3
0,50	80	30	12	4
0,55	120	50	16	6
0,60	160	60	24	8
0,65	180	90	32	12
0,70	220	130	48	18
0,75	300	180	80	28
0,80	360	240	124	46
0,85	460	320	196	86
0,90	620	450	308	164

3328×1664, cinco bandas, bandas polares, QP37

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	4	2	1	1
0,15	6	2	1	1
0,20	10	3	2	1
0,25	14	4	2	1
0,30	21	6	3	2
0,35	32	10	4	2
0,40	47	15	5	3
0,45	70	23	8	4
0,50	100	40	12	4
0,55	140	50	16	6
0,60	180	70	24	8
0,65	200	100	36	12
0,70	280	140	52	18
0,75	340	200	84	28
0,80	400	260	136	50
0,85	520	350	208	90
0,90	660	480	324	172

3328×1664, cinco bandas, bandas polares, outro QP

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	3	2	1	1
0,15	5	2	1	1
0,20	8	3	1	1
0,25	11	4	2	1
0,30	15	5	2	1
0,35	21	7	3	2
0,40	31	11	5	2
0,45	45	17	6	3
0,50	80	30	12	4
0,55	100	40	20	6
0,60	140	50	28	8
0,65	160	70	28	10
0,70	200	100	40	16
0,75	260	140	60	24
0,80	340	200	100	38
0,85	420	290	164	70
0,90	588	410	268	142

3328×1664, cinco bandas, bandas meio-polares, QP22

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	2	2	1	1
0,15	4	2	1	1
0,20	7	3	2	1
0,25	11	4	2	1
0,30	16	5	3	2
0,35	22	9	5	3
0,40	33	13	7	4
0,45	46	19	9	5
0,50	60	30	12	6
0,55	80	40	20	8
0,60	100	50	24	10
0,65	120	70	36	14
0,70	160	90	48	22
0,75	200	130	76	32
0,80	280	190	116	54
0,85	380	280	200	106
0,90	540	460	368	226

3328×1664, cinco bandas, bandas meio-polares, QP27

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	5	2	1	1
0,15	9	3	2	1
0,20	15	5	2	1
0,25	25	7	3	2
0,30	37	11	5	2
0,35	51	17	7	3
0,40	65	25	10	4
0,45	85	35	14	6
0,50	120	50	20	8
0,55	140	70	28	10
0,60	180	90	40	14
0,65	240	130	60	20
0,70	320	180	88	32
0,75	400	260	140	52
0,80	480	380	224	92
0,85	700	540	368	178
0,90	1000	820	596	344

3328×1664, cinco bandas, bandas meio-polares, QP32

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	5	2	1	1
0,15	10	3	2	1
0,20	17	5	2	1
0,25	30	8	3	2
0,30	43	13	5	2
0,35	58	20	8	3
0,40	78	29	11	4
0,45	101	41	15	6
0,50	140	60	24	8
0,55	180	80	32	12
0,60	220	110	44	16
0,65	300	160	68	22
0,70	380	220	104	34
0,75	460	320	164	58
0,80	600	450	264	106
0,85	820	640	412	200
0,90	1140	920	656	378

3328×1664, cinco bandas, bandas meio-polares, QP37

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	3	1	1	1
0,10	5	2	1	1
0,15	11	3	2	1
0,20	19	5	2	1
0,25	33	9	3	2
0,30	48	14	5	2
0,35	64	22	8	3
0,40	86	32	11	5
0,45	116	45	16	6
0,50	160	70	24	8
0,55	200	90	36	12
0,60	260	130	48	16
0,65	340	180	72	24
0,70	420	260	112	36
0,75	520	360	180	62
0,80	720	490	288	112
0,85	940	700	440	214
0,90	1240	990	696	396

3328×1664, cinco bandas, bandas meio-polares, outro QP

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	2	1	1	1
0,10	5	2	1	1
0,15	8	3	2	1
0,20	15	4	2	1
0,25	25	7	3	2
0,30	38	11	5	2
0,35	52	17	7	3
0,40	70	40	10	4
0,45	90	36	14	6
0,50	120	50	20	8
0,55	160	70	28	10
0,60	200	100	40	14
0,65	260	140	56	20
0,70	360	190	88	30
0,75	420	280	136	50
0,80	540	400	224	90
0,85	780	570	372	174
0,90	1080	860	596	340

4096×2048, cinco bandas, bandas polares, QP22

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	1	1	1	1
0,20	1	1	1	1
0,25	2	1	1	1
0,30	2	1	1	1
0,35	2	1	1	1
0,40	2	1	1	1
0,45	2	2	1	1
0,50	2	2	2	1
0,55	2	2	2	1
0,60	2	2	2	2
0,65	3	2	2	2
0,70	3	2	2	2
0,75	4	2	2	2
0,80	5	3	2	2
0,85	6	3	2	3
0,90	9	4	2	3

4096×2048, cinco bandas, bandas polares, QP27

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	1	1	1	1
0,20	2	1	1	1
0,25	2	1	1	1
0,30	2	1	1	1
0,35	2	2	1	1
0,40	2	2	2	2
0,45	2	2	2	2
0,50	2	2	2	2
0,55	3	2	2	2
0,60	3	2	2	2
0,65	3	2	2	3
0,70	4	2	2	4
0,75	5	3	3	5
0,80	6	3	4	7
0,85	8	4	7	9
0,90	9	4	9	12

4096×2048, cinco bandas, bandas polares, QP32

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	1	1	1	1
0,20	2	1	1	1
0,25	2	2	1	1
0,30	2	2	2	1
0,35	2	2	2	2
0,40	2	2	2	2
0,45	2	2	2	2
0,50	3	2	2	2
0,55	3	3	3	3
0,60	4	3	5	4
0,65	5	4	6	5
0,70	6	6	8	6
0,75	7	9	10	8
0,80	9	11	12	10
0,85	11	14	15	13
0,90	15	17	19	17

4096×2048, cinco bandas, bandas polares, QP37

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	2	1	1	1
0,15	2	1	1	1
0,20	2	2	1	1
0,25	2	2	2	1
0,30	2	2	2	1
0,35	3	2	2	2
0,40	4	3	2	2
0,45	6	3	2	2
0,50	8	4	3	2
0,55	10	7	5	3
0,60	12	9	7	4
0,65	14	11	8	5
0,70	15	13	10	7
0,75	17	15	12	8
0,80	19	17	15	10
0,85	21	20	18	13
0,90	24	23	22	18

4096×2048, cinco bandas, bandas polares, outro QP

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	2	1	1	1
0,20	2	1	1	1
0,25	2	1	1	1
0,30	2	2	2	1
0,35	2	2	2	1
0,40	2	2	2	2
0,45	3	2	2	2
0,50	3	2	2	2
0,55	4	3	2	2
0,60	5	3	2	2
0,65	6	4	3	3
0,70	8	5	5	4
0,75	10	8	8	6
0,80	13	11	10	8
0,85	16	14	13	10
0,90	19	18	17	14

4096×2048, cinco bandas, bandas meio-polares, QP22

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	1	1	1	1
0,20	2	1	1	1
0,25	2	1	1	1
0,30	2	1	1	1
0,35	2	1	1	1
0,40	2	2	1	1
0,45	3	2	2	1
0,50	3	2	2	2
0,55	3	2	2	2
0,60	4	2	2	2
0,65	4	2	2	2
0,70	5	3	3	3
0,75	6	4	3	3
0,80	7	5	5	4
0,85	11	8	6	6
0,90	22	15	9	8

4096×2048, cinco bandas, bandas meio-polares, QP27

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	1	1	1	1
0,20	2	1	1	1
0,25	2	1	1	1
0,30	2	2	1	1
0,35	2	2	2	1
0,40	3	2	2	2
0,45	3	2	2	2
0,50	4	2	2	2
0,55	4	3	3	3
0,60	5	4	4	3
0,65	6	5	5	4
0,70	7	6	6	5
0,75	8	8	8	6
0,80	11	10	9	8
0,85	17	13	12	10
0,90	35	20	16	14

4096×2048, cinco bandas, bandas meio-polares, QP32

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	2	1	1	1
0,20	2	1	1	1
0,25	2	2	1	1
0,30	3	2	2	1
0,35	3	2	2	1
0,40	4	3	2	2
0,45	5	3	3	2
0,50	6	4	4	2
0,55	6	6	5	3
0,60	8	7	6	4
0,65	9	8	7	5
0,70	11	10	9	6
0,75	13	12	10	7
0,80	16	15	13	9
0,85	21	18	16	12
0,90	38	26	21	16

4096×2048, cinco bandas, bandas meio-polares, QP37

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	2	1	1	1
0,15	2	1	1	1
0,20	2	2	1	1
0,25	3	2	1	1
0,30	4	2	2	1
0,35	5	3	2	2
0,40	6	4	2	2
0,45	7	5	3	2
0,50	8	6	4	3
0,55	10	7	5	3
0,60	11	9	7	4
0,65	13	11	8	5
0,70	15	13	10	6
0,75	17	15	12	8
0,80	20	17	14	10
0,85	26	22	17	12
0,90	43	30	23	17

4096×2048, cinco bandas, bandas meio-polares, outro QP

<b><i>Cutoff</i></b>	<b>Tamanho de CU</b>			
	<b>64×64</b>	<b>32×32</b>	<b>16×16</b>	<b>8×8</b>
0,05	1	1	1	1
0,10	1	1	1	1
0,15	2	1	1	1
0,20	2	1	1	1
0,25	2	2	1	1
0,30	3	2	1	1
0,35	3	2	2	1
0,40	4	2	2	2
0,45	4	3	2	2
0,50	5	3	3	2
0,55	6	5	4	3
0,60	7	6	5	3
0,65	9	7	6	4
0,70	11	9	7	5
0,75	14	11	9	6
0,80	17	14	11	8
0,85	22	18	14	10
0,90	36	25	19	14

## **APÊNDICE B DISTRIBUIÇÃO DOS TAMANHOS DE PU DE ACORDO COM A LINHA DE CTUS**



Distribuição dos tamanhos de PU de acordo com a linha de CTUs para vídeos codificados na resolução  $3328 \times 1664$

Linha de CTUs												
PU	0	1	2	3	4	5	6	7	8	9	10	11
<b>4×4</b>	0,00089	0,00485	0,01148	0,02041	0,02728	0,03603	0,04556	0,06039	0,06933	0,08140	0,09397	0,11052
<b>8×8</b>	0,01114	0,03060	0,04941	0,06926	0,08207	0,09447	0,11056	0,13460	0,15116	0,16595	0,17502	0,19666
<b>16×16</b>	0,06930	0,12160	0,14748	0,17067	0,17908	0,18867	0,21898	0,23400	0,25404	0,26368	0,26555	0,27490
<b>32×32</b>	0,41501	0,39481	0,36470	0,35335	0,33914	0,34596	0,36163	0,33374	0,32823	0,32313	0,31602	0,29217
<b>64×64</b>	0,50366	0,44814	0,42692	0,38631	0,37243	0,33486	0,26327	0,23726	0,19724	0,16584	0,14944	0,12574
Linha de CTUs												
PU	12	13	14	15	16	17	18	19	20	21	22	23
<b>4×4</b>	0,13402	0,14268	0,12063	0,09837	0,08773	0,07808	0,06874	0,05921	0,05286	0,04142	0,02874	0,01868
<b>8×8</b>	0,22638	0,23257	0,21163	0,18803	0,17775	0,17380	0,15963	0,14339	0,14144	0,12915	0,10554	0,07909
<b>16×16</b>	0,28128	0,27035	0,27660	0,26688	0,25745	0,26032	0,25579	0,25004	0,25591	0,26896	0,26059	0,22717
<b>32×32</b>	0,25280	0,24752	0,26610	0,29110	0,28993	0,29125	0,30225	0,32179	0,32210	0,34492	0,38061	0,40623
<b>64×64</b>	0,10552	0,10687	0,12505	0,15562	0,18714	0,19656	0,21360	0,22557	0,22768	0,21555	0,22452	0,26883
Linha de CTUs												
PU	24	25										
<b>4×4</b>	0,00783	0,00088										
<b>8×8</b>	0,04754	0,01495										
<b>16×16</b>	0,18415	0,11230										
<b>32×32</b>	0,43755	0,52157										
<b>64×64</b>	0,32292	0,35029										

Distribuição dos tamanhos de PU de acordo com a linha de CTUs para vídeos codificados na resolução  $4096 \times 2048$

Linha de CTUs												
PU	0	1	2	3	4	5	6	7	8	9	10	11
<b>4×4</b>	0,00089	0,00392	0,00851	0,01527	0,02206	0,02753	0,03462	0,04157	0,05232	0,06313	0,06997	0,07974
<b>8×8</b>	0,01114	0,02705	0,04186	0,05814	0,07271	0,08245	0,09237	0,10380	0,12139	0,13958	0,15217	0,16444
<b>16×16</b>	0,06930	0,11462	0,13837	0,15803	0,17388	0,17919	0,18548	0,20852	0,22693	0,23870	0,25513	0,26319
<b>32×32</b>	0,41501	0,40091	0,37462	0,35884	0,35014	0,33891	0,34294	0,36011	0,35105	0,32965	0,32814	0,32393
<b>64×64</b>	0,50366	0,45349	0,43664	0,40973	0,38120	0,37192	0,34459	0,28600	0,24830	0,22893	0,19459	0,16871
Linha de CTUs												
PU	12	13	14	15	16	17	18	19	20	21	22	23
<b>4×4</b>	0,08983	0,10096	0,11721	0,13596	0,14330	0,12860	0,10769	0,09407	0,08655	0,07870	0,07122	0,06318
<b>8×8</b>	0,17165	0,18337	0,20577	0,22836	0,23338	0,21918	0,19834	0,18320	0,17726	0,17427	0,16427	0,14931
<b>16×16</b>	0,26462	0,26913	0,27832	0,28055	0,27079	0,27494	0,27355	0,26255	0,25741	0,26022	0,25768	0,25181
<b>32×32</b>	0,31917	0,30764	0,28021	0,25048	0,24666	0,25892	0,28092	0,29292	0,28949	0,29091	0,29793	0,31447
<b>64×64</b>	0,15472	0,13889	0,11849	0,10464	0,10587	0,11836	0,13950	0,16726	0,18928	0,19591	0,20890	0,22123
Linha de CTUs												
PU	24	25	26	27	28	29	30	31				
<b>4×4</b>	0,05696	0,05143	0,04184	0,03138	0,02283	0,01448	0,00596	0,00088				
<b>8×8</b>	0,14195	0,14059	0,12978	0,11127	0,09062	0,06742	0,04111	0,01495				
<b>16×16</b>	0,25050	0,25816	0,26875	0,26507	0,24292	0,21207	0,17340	0,11230				
<b>32×32</b>	0,32257	0,32370	0,34382	0,37302	0,39655	0,41547	0,44807	0,52157				
<b>64×64</b>	0,22802	0,22612	0,21580	0,21926	0,24707	0,29056	0,33146	0,35029				

## **ANEXO A PRIMEIRO QUADRO DAS SEQUÊNCIAS OMNIDIRECIONAIS UTILIZADAS**



**AerialCity**



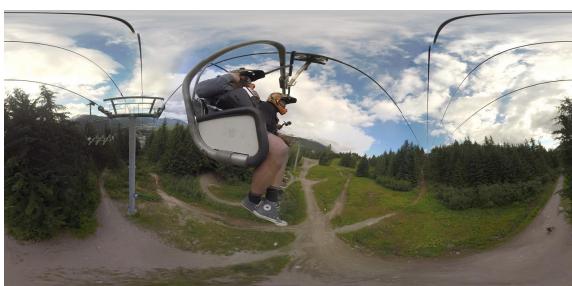
**Balboa**



**BranCastle**



**Broadway**



**ChairliftRide**



**GasLamp**



Harbor



KiteFlite



Landing



PoleVault



SkateboardInLot



Trolley