

AVALIAÇÃO SUBJETIVA DA QUALIDADE DO DMMFAST NO PADRÃO EMERGENTE 3D-HEVC

MÁRIO SALDANHA; GABRIEL BALOTA; GUSTAVO SANCHEZ;
BRUNO ZATT; LUCIANO AGOSTINI; MARCELO PORTO

Universidade Federal de Pelotas
{*mrdfsaldanha, gmbalota, gfsanchez, zatt, agostini, porto*}@inf.ufpel.edu.br
Grupo de Arquiteturas e Circuitos Integrados

1. INTRODUÇÃO

Aplicações 3D estão se tornando cada vez mais populares, com isso, cresce o interesse em pesquisar e desenvolver técnicas para reduzir a quantidade de dados necessários para transmitir ou reproduzir vídeos 3D. Vídeos 3D geram uma quantidade de dados consideravelmente maior quando comparados a vídeos 2D, devido as múltiplas câmeras necessárias para capturar o vídeo. Este volume de dados aumenta ainda mais na codificação de vídeos de alta definição.

A codificação de mapas de profundidade é uma das técnicas utilizada no padrão emergente 3D-*High Efficiency Video Coding* (3D-HEVC) (TECH, 2013) para codificação de vídeos 3D. Os mapas de profundidade são imagens em tons de cinza que possuem uma representação com 8 bits por amostra, variando do valor 0 (preto) até o valor 255 (branco). Cada quadro de textura está associado a um mapa de profundidade correspondente, indicando a proximidade dos objetos em relação à câmera. Objetos mais próximos da câmera possuem seus valores próximos de 255, enquanto objetos mais distantes da câmera possuem valores próximos de 0. Assim, é possível representar os objetos próximos da câmera de maneira mais clara, enquanto objetos mais distantes são representados de maneira mais escura. A vantagem da utilização de mapas de profundidade associados a quadros de textura é que é possível gerar, eficientemente, vídeos de pontos de vista intermediários através da interpolação de canais de textura e de profundidade (MERKLE, 2007).

Os mapas de profundidade possuem duas características importantes, uma grande quantidade de regiões homogêneas e arestas bem definidas, possibilitando identificar facilmente os objetos presentes no quadro. Os algoritmos de codificação da predição intra-quadro tradicionais utilizados na codificação de vídeos 2D podem ser usados para codificação dos mapas de profundidade, contudo, estes algoritmos podem não alcançar uma boa eficiência na codificação de blocos com arestas, podendo gerar artefatos nas vistas intermediárias sintetizadas (MULLER, 2011). Com base neste fato, foram desenvolvidos quatro novos modos para predição intra-quadro para codificação de profundidade, estes são conhecidos como modos de modelagem de profundidade (DMM – *Depth Modeling Modes*). Esses modos consistem em dividir o bloco que está sendo codificado em duas regiões e representam cada região por um valor constante.

O modo 1 (DMM 1) é o modo que apresenta a maior complexidade comparado aos outros 3 modos, este modo realiza a divisão do bloco através de uma partição *Wedgelet* apresentada na Figura 1. Na partição *Wedgelet*, duas regiões (P_1 e P_2) são separadas por uma linha reta. Porém, a escolha da reta é feita exaustivamente em uma busca sobre o conjunto de partições *Wedgelets*. A partição *Wedgelet* que alcançar o menor erro de codificação (distorção) entre o sinal original e a predição através da *Wedgelet* é selecionada.

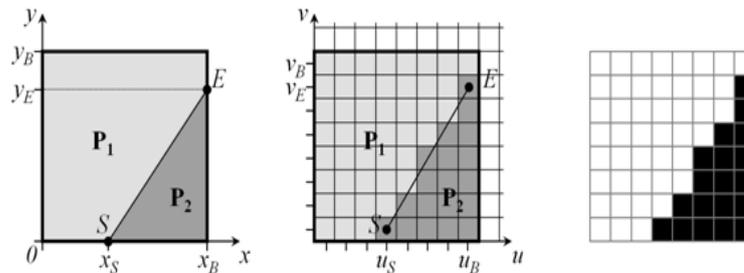


Figura 1 – Predição de profundidade por *Wedgelet* (MERKLE, 2011).

Este trabalho tem como objetivo apresentar uma avaliação subjetiva de qualidade, analisando a combinação, chamada de DMMFast (*Depth Modeling Modes Fast Prediction*), de dois algoritmos desenvolvidos: *Simplified Edge Detection* (SED) e *Gradient-based Mode One Filter* (GMOF). O algoritmo SED classifica o bloco de profundidade que está sendo codificado como aresta ou região constante, com base nisso é decidido se é realizada a codificação dos modos DMM ou apenas a codificação dos modos intra-quadro tradicionais. O algoritmo GMOF foi desenvolvido com o objetivo de reduzir a complexidade do algoritmo DMM 1, reduzindo o número de *Wedgelets* avaliadas com base nos N maiores gradientes encontrados nas bordas do bloco que está sendo codificado.

2. METODOLOGIA

O DMMFast é composto de dois algoritmos: SED e GMOF. O algoritmo SED tem como objetivo decidir se é necessária a avaliação dos modos DMM (que foram desenvolvidos principalmente para codificação de arestas) ou apenas a avaliação dos modos intra-quadro tradicionais durante a codificação do bloco. O algoritmo computa os valores dos quatro cantos do bloco (superior esquerda, superior direita, inferior esquerda e inferior direita), representados na Figura 2(a) por A, B, C e D. Então, realiza a diferença absoluta desses valores, a maior diferença é comparada com um limiar (definido com base em uma análise estatística), que se adapta de acordo com o tamanho do bloco e a resolução do vídeo. Se a maior diferença for maior que o limiar definido, o bloco é classificado como aresta, e então não é realizada nenhuma simplificação na codificação, pois os modos DMM são adicionados na lista para avaliação como pode ser observado na Figura 2(b). Caso contrário, o bloco é classificado como uma região constante, neste caso os modos DMM não são adicionados na lista, ou seja, não são avaliados durante a etapa de codificação, realizando a avaliação apenas dos modos intra-quadro tradicionais.

O algoritmo GMOF tem como objetivo reduzir o número de partições *Wedgelets* que são avaliadas durante a execução do DMM 1. Os gradientes da borda superior, inferior, esquerda e direita são computados realizando a diferença absoluta entre pixels vizinhos. Os gradientes são armazenados em uma lista em ordem decrescente. Após esta avaliação, as primeiras N posições da lista armazenam os maiores gradientes, então somente as *Wedgelets* que iniciam (ou terminam) nas primeiras N posições são avaliadas, evitando uma avaliação de todo conjunto de partições *Wedgelets*. O parâmetro N utilizado para essa avaliação foi igual a 8. Foi definido após a realização de algumas simulações em software, notamos que o valor escolhido alcançou um bom equilíbrio em complexidade computacional e qualidade de vídeo.

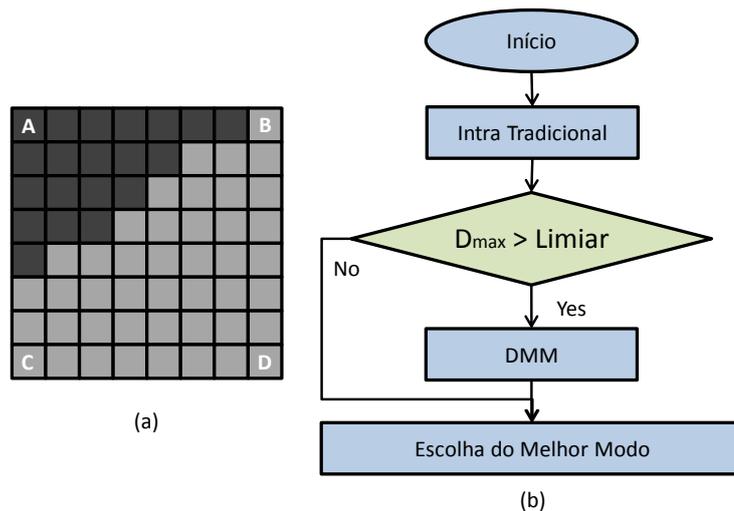


Figura 2 – (a) Bloco 8x8. (b) Diagrama de bloco do algoritmo SED.

3. RESULTADOS E DISCUSSÃO

Uma análise subjetiva foi realizada para avaliar os resultados de qualidade do DMMFast. Para realizar a avaliação subjetiva foi utilizado o *Degradation Category Rating* (DCR) (ITU, 2009), onde o vídeo original é apresentado, e após um intervalo de 2 segundos, é apresentado o vídeo codificado com os algoritmos que estão sendo avaliados. Para realizar o experimento foram utilizados os vídeos presentes nas condições comuns de teste (CCT) e foram apresentados para 26 estudantes de Engenharia de Computação, com idade variando de 17 até 22 anos, sem nenhum conhecimento em codificação de vídeo. Primeiramente foram comparados os vídeos originais com os vídeos codificados pelo software de referência 3D-HEVC *Test Model 7.0* (HTM) (TECH, 2013), e então os estudantes atribuíram uma nota na escala de 1 até 5 pontos. Após, foram comparados os vídeos originais com os vídeos codificados com o DMMFast, e os participantes também atribuíram notas considerando a mesma escala.

Analisando a Tabela 1 observamos que o HTM 7.0 alcançou um ganho em média de 0,0261 pontos (na escala de 1 até 5) nos resultados de qualidade, quando comparado ao DMMFast. Este resultado representa praticamente nenhuma degradação quando a qualidade de vídeo é avaliada. Contudo, o DMMFast foi capaz de alcançar em média um ganho no bit rate de 0,371% ao realizar a avaliação objetiva.

A degradação na qualidade do vídeo foi notada pelos participantes em média principalmente nos vídeos de alta definição (1920x1088). Nos vídeos de baixa definição (1024x768) em média o DMMFast alcançou um resultado melhor que o HTM 7.0.

Com esta análise podemos notar que a solução proposta apresenta uma pequena degradação na qualidade, quando comparado ao software de referência HTM 7.0, e além disso, é capaz de fornecer uma importante redução na complexidade do codificador, em média de 6,3%, e reduzir o bit rate em 0,371% em média.

4. CONCLUSÕES

Este trabalho teve como objetivo realizar uma análise subjetiva da qualidade

de uma combinação de dois algoritmos desenvolvidos com objetivo de reduzir a complexidade da codificação mapas de de profundidade em vídeo 3D. Esta estratégia foi chamada de DMMFast, e utiliza os algoritmos SED (*Simplified Edge Detection*) e GMOF (*Gradient-based Mode One Filter*). O algoritmo SED evita a execução dos novos modos para codificação de profundidade quando um bloco é classificado como aresta, reduzindo consideravelmente a complexidade. O algoritmo GMOF realiza uma simplificação mais específica no DMM 1, reduzindo o número de partições *Wedgelets* necessárias para avaliação, baseado nos gradientes das bordas do bloco.

Com a análise subjetiva da qualidade foi possível concluir que ao utilizar o DMMFast foi possível obter uma redução considerável na redução da complexidade do codificador, 6,3% em média, além de uma redução média do bit rate de 0,371%, com uma degradação na qualidade do vídeo de apenas 0,0261 em média, considerando a escala de 1 até 5 pontos.

Tabela 1 – Resultados da análise subjetiva do DMMFast.

Resolução	Vídeos	Diferença Média (Pontos)	Bit rate	Tempo de codificação
1024x768	Balloons	-0,0326	-0.298%	94,2%
	Kendo	0,0231	-0.181%	94,3%
	Newspaper_CC	-0,5192	-1.111%	93,5%
1920x1088	GT_Fly	0,0538	-0.210%	93,8%
	Poznan_Hall2	0,8000	-0.204%	93,7%
	Poznan_Street	0,0154	-0.610%	92,6%
	Undo_Dancer	-0,1576	0.020%	94,0%
Média	1024x768	-0,1762	-0.005%	94,0%
	1920x1088	0,1779	-0.002%	93,5%
	Média	0,0261	-0.371%	93,7%

5. REFERÊNCIAS BIBLIOGRÁFICAS

TECH, G.; WEGNER, K.; CHEN, Y.; YEA, S. **3D HEVC Test Model 3.**

Documento: JCT3V-C1005. Draft 3 of 3D-HEVC Test Model Description. Geneva, 2013.

MERKLE, P.; et al. **3D video: Depth Coding Based on Inter-Component Prediction of Block Partitions.** IEEE Picture Coding Symposium (PCS), 2011.

MULLER, K. et al. 3D High-Efficiency Video Coding for Multi-View Video and Depth Data. **IEEE Transactions on Image Processing**, V. 22, n. 9, p. 3366 – 3378, 2013.

ITU – **International Telecommunication Union** “Subjective Video Quality Assessment Methods for Multimedia Applications”, p. 910, Feb. 2009.