

## ARQUITETURA PARA O ALGORITMO DE DECISÃO GREEDY AND FLEETING STEP NA PREDIÇÃO INTRA-QUADRO PARA MAPAS DE PROFUNDIDADE

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

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

### 1. INTRODUÇÃO

Atualmente a diversidade de dispositivos que suportam aplicações 3D vem crescendo no mercado mundial. Vídeos 3D possuem uma enorme quantidade de dados, principalmente em vídeos de alta definição (HD), que precisam ser processados, transmitidos e/ou armazenados. Por exemplo, em um vídeo 3D com duração de 10 minutos e resolução HD 1080p a uma taxa de 30 quadros por segundo, com cinco vistas de textura sem compressão, seriam necessários 50 GB para armazená-lo. Com a adição de canais de profundidade o volume de dados cresce em 33,3%. Dado o grande volume de dados, os dispositivos que processam tais vídeos devem ter grande capacidade de processamento. Adicionalmente, como muitos dos dispositivos multimídia atuais são móveis, é necessário primar pelo baixo consumo de potência. Portanto, arquiteturas de hardware dedicadas são necessárias devido a seu alto desempenho e baixo consumo.

O padrão emergente *3D-High Efficiency Video Coding (3DVC)* é uma extensão do padrão de codificação de vídeo *High Efficiency Video Coding (HEVC)* (ITU-T, 2013). Para codificar os vídeos 3D, algoritmos de codificação 2D não conseguem atingir uma grande eficiência por não explorarem as redundâncias entre vistas. Com isso, novos algoritmos são explorados para atingir maior eficiência de codificação de vídeos 3D de alta definição.

Mapas de profundidade trazem informação adicional aos vídeos tradicionais (textura) e sua codificação está inserida no padrão emergente 3DVC. Com isso, cada canal de textura deve estar associado a um canal de profundidade (MULLER, 2013). Estes canais de profundidade são vídeos monocromáticos que indicam a distância entre objetos e a câmera. Estes mapas de profundidade possuem duas características bem distintas dos quadros de textura: regiões muito homogêneas e arestas bem definidas. Arestas são codificadas de forma ineficiente pela predição intra tradicional e em regiões homogêneas a predição intra gera ótimos resultados. Para contornar o fato da predição intra ser ineficiente em arestas foram desenvolvidos novos modos de predição intra que devem ser aplicados apenas em quadros de profundidade.

O algoritmo *Greedy and Fleeting Step (GFS)* foi desenvolvido com o objetivo de reduzir a complexidade da predição intra-quadros de mapas de profundidade. O algoritmo determina se um bloco deve ser codificado como se ele fosse uma aresta ou uma região homogênea. No caso de ser uma aresta, a etapa da codificação intra tradicional é descartada e são testados apenas os novos modos de predição, caso contrário, são avaliados todos os modos. O GFS faz uma comparação entre os quatro pixels dos extremos do bloco para decidir se a região do bloco que está sendo codificada é homogênea ou aresta. Estes quatro pixels são representativos, pois em geral uma aresta é separada por uma reta com um gradiente abrupto.

Este artigo apresenta o projeto de uma arquitetura em hardware para o algoritmo GFS. Esta arquitetura é responsável pela tomada de decisão entre utilizar somente os novos modos intra-quadro ou utilizar os modos tradicionais. A arquitetura proposta suporta blocos de tamanho 4x4, 8x8, 16x16 e 32x32.

## 2. METODOLOGIA

A arquitetura foi descrita na linguagem VHDL, onde um bloco do mapa de profundidade de tamanho 32x32 é lido linha a linha (os outros tamanhos de bloco estão contidos dentro dele), onde cada linha possui 32 bytes. A arquitetura possui registradores de entrada, onde são armazenadas as informações que são relevantes para a tomada de decisão de cada tamanho de bloco, conforme mostrado na Tabela 1.

Tabela 1 – Informações para os registradores de entrada

Tamanho de bloco	Pixels Relevantes
4x4	1, 4, 5, 8, 9, 12, 13, 16, 17, 20, 21, 24, 25, 28, 29 e 32
8x8	1, 8, 9, 16, 17, 24, 25 e 32
16x16	1, 16, 17 e 32
32x32	1 e 32

A Figura 1 apresenta o diagrama de blocos para o algoritmo. No momento em que algum bloco de qualquer tamanho estiver pronto, a entrada é entregue diretamente ao núcleo de processamento para que seja feita a comparação, como é apresentado na Figura 2. Este núcleo é responsável por calcular a maior diferença entre os quatro pixels dos cantos e determinar se os modos de previsão intra tradicionais devem ser aplicados. Após utilizar as informações dos bancos de registradores da entrada, no próximo ciclo estas são descartadas para que novas informações sejam armazenadas.

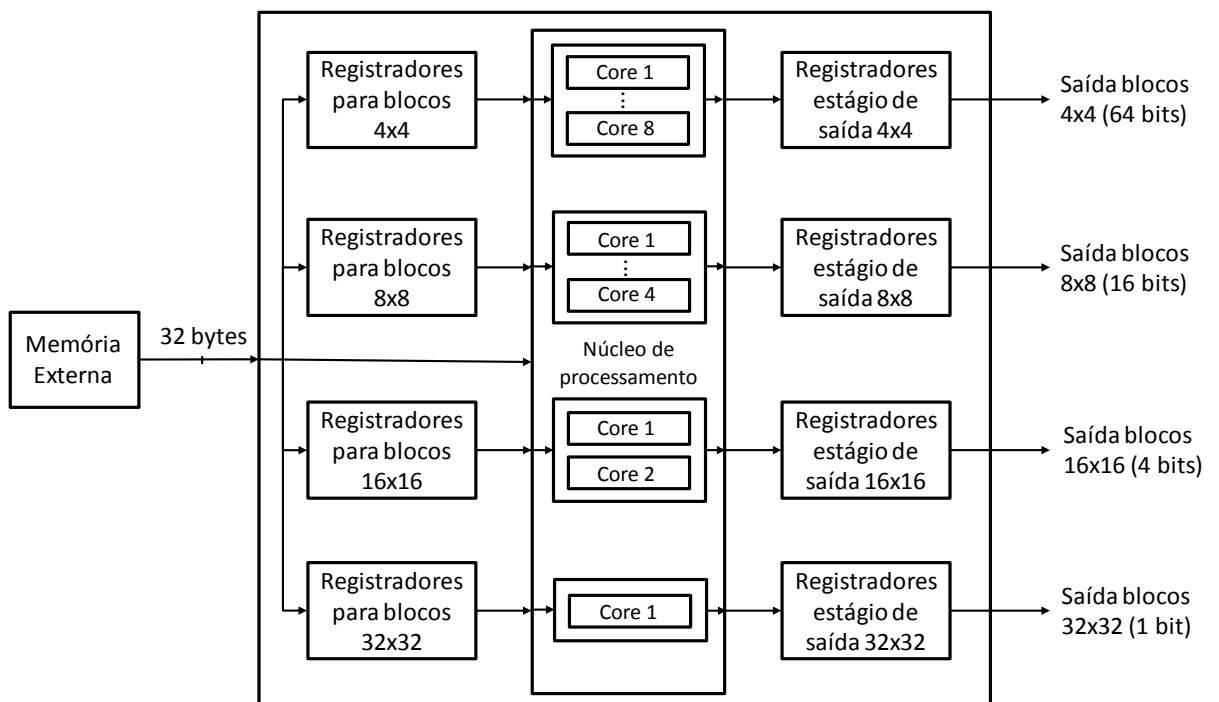


Figura 1 – Design arquitetura GFS

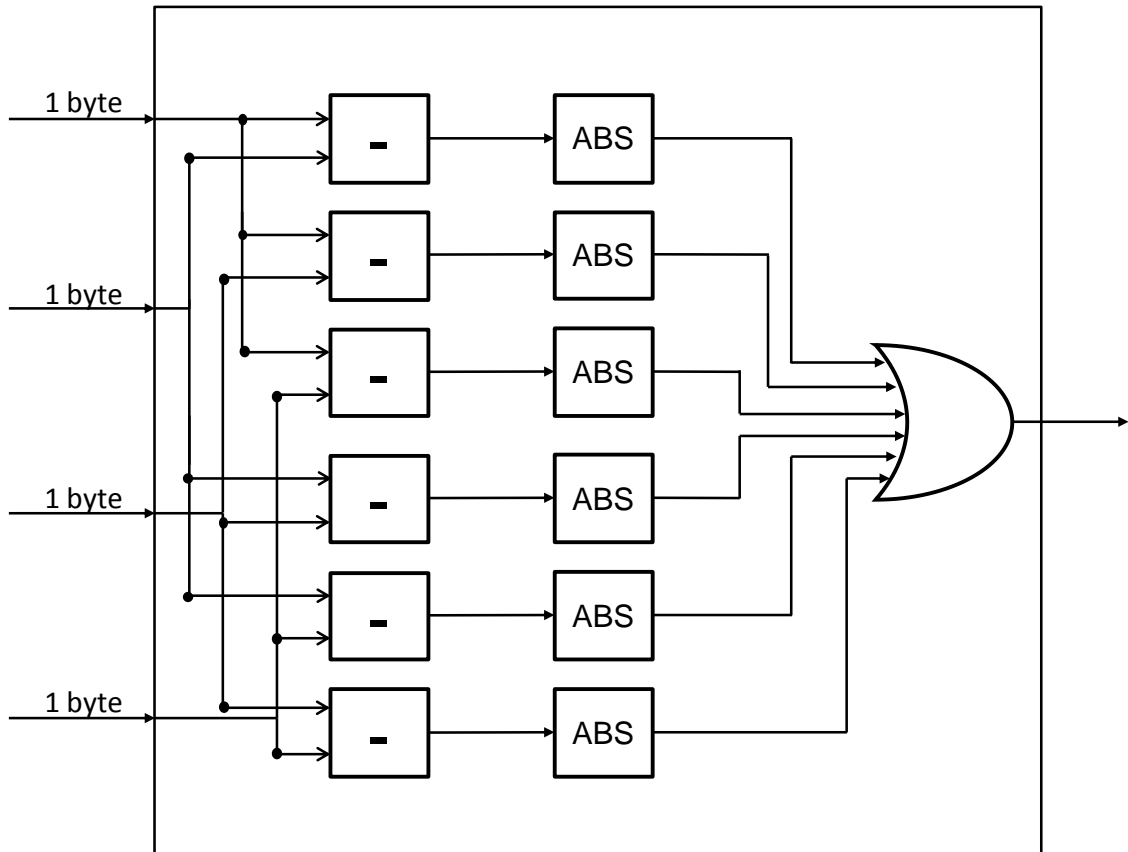


Figura 2 – Core interno ao Núcleo de Processamento da arquitetura GFS

O núcleo de processamento possui 15 *cores*, onde executa a comparação entre a subtração dos pixels dos extremos de um bloco e uma constante. São oito *cores* para blocos 4x4, quatro *cores* para bloco 8x8, dois *cores* para blocos 16x16 e um *core* para blocos 32x32. A execução desses *cores* é feita de acordo com o tamanho de bloco. Se a constante for maior que o valor da subtração, a codificação executará somente novos modos de predição intra-quadro. Se a subtração for menor que a constante, além dos novos modos de predição, executará os modos de predição intra-quadros tradicionais.

A arquitetura GFS necessita de 32 ciclos para ler um bloco completo 32x32. Para processar um bloco de profundidade 32x32 completo é necessário mais um ciclo para que as informações sobre a decisão dos modos de predição sejam processadas.

### 3. RESULTADOS E DISCUSSÃO

A arquitetura foi sintetizada para a tecnologia ASIC 90nm para duas frequências diferentes, cada uma responsável por obter uma taxa de processamento suficiente para a codificação de vídeos em tempo real. As resoluções alvos foram: HD 1080p e *Quad Full HD* (QFHD).

É possível notar que em um vídeo de 1920x1080 pixel existe 2.025 blocos de tamanho 32x32, ou seja, em um quadro de profundidade é necessário de 66.825 ciclos para completar seu processamento na arquitetura. Com base nessas informações é necessário que a arquitetura processe os dados a frequências de 2,005 MHz e 8,02 MHz, para vídeos HD 1080p e QFHD, respectivamente.

Além disso, por ser um vídeo 3D é necessário codificar várias vistas ao mesmo tempo. Se um vídeo com uma resolução de 1080p possuir cinco vistas,

para processar todos os blocos dessas cinco vistas será necessário de 334.125 ciclos. Para a arquitetura processar as cinco vistas, a frequência é multiplicada por cinco, ou seja, a arquitetura para esse vídeo teria uma frequência de 10,025 MHz. Na Tabela 2 são apresentados os resultados de síntese para cada frequência em um vídeo com cinco vistas e com resoluções de 1080p e QFHD.

Tabela 2 – Resultado de síntese para tecnologia 90nm

Resolução Alvo	Frequência (MHz)	Área ( $\mu\text{m}$ )	Potência dinâmica ( $\mu\text{W}$ )	Potência Leakage ( $\mu\text{W}$ )
HD 1080p	10,025	23,181	111,6377	53,1407
QFHD	40,1	23,192	100,8558	53,2210

#### 4. CONCLUSÕES

Os mapas de profundidade são formados por imagens monocromáticas e possuem arestas bem definidas e regiões bastante homogêneas. Blocos que contenham arestas são codificados de forma ineficiente pela predição intra-tradicional e ao determinar se um bloco representa uma aresta pode ser feita uma simplificação na codificação do vídeo.

Este trabalho apresentou uma arquitetura para o algoritmo GFS que tem o objetivo de diminuir a complexidade na codificação intra-quadros de mapas de profundidade. A idéia principal deste algoritmo é determinar se a predição intra-quadros tradicional será utilizada em função de características internas ao bloco que está sendo codificado. Resultados de síntese demonstram que a arquitetura desenvolvida para este algoritmo necessita de poucos recursos de hardware e apresenta um baixo consumo de potência.

#### 5. REFERÊNCIAS BIBLIOGRÁFICAS

RICHARDSON, I. **The H.264 Advanced Video Compression Standard. 2** John Wiley and Sons, 2010.

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.

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-T. **SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS. Infrastructure of audiovisual services – Coding of moving vídeo.** High efficiency video coding 2013