

HARDWARE DEDICADO PARA A ETAPA DE ESTIMAÇÃO DE MOVIMENTO DO PADRÃO HEVC VISANDO O PROCESSAMENTO DE VÍDEOS UHD

MURILO PERLEBERG¹; VLADIMIR AFONSO²;
MARCELO PORTO¹; BRUNO ZATT¹

¹*Universidade Federal de Pelotas – Video Technology Research Group (ViTech)*

²*Instituto Federal Sul-rio-grandense – Campus Pelotas*

{mrperleberg, vafonso, porto, zatt}@inf.ufpel.edu.br

1. INTRODUÇÃO

Atualmente a popularidade dos vídeos digitais é crescente, principalmente devido à possibilidade de *streaming* através da internet e à melhoria da capacidade de reprodução destes vídeos por dispositivos móveis, como *smartphones* e *tablets*. Porém, como vídeos digitais requerem uma grande quantidade de dados para serem representados e, consequentemente, armazenados e transmitidos, o uso de técnicas de compressão são obrigatórias.

Um codificador de vídeo é composto de várias etapas responsáveis por explorar as redundâncias presentes nos dados em conformidade com um padrão de codificação de vídeos. Dentre os padrões disponíveis, o *High Efficiency Video Coding* (HEVC) (ITU-T, 2013) é considerado o estado da arte (OHM, 2012). A principal etapa de um codificador baseado no padrão HEVC é a Estimação de Movimento (ME – *Motion Estimation*) (AFONSO, 2016), que é dividida em dois estágios principais: a ME Inteira (IME – *Integer ME*), que aplica um algoritmo de busca para encontrar uma região similar a que está sendo codificada em quadros previamente processados, e a ME Fracionária (FME – *Fractional ME*), que realiza um refinamento no resultado obtido pela IME.

O processo de codificação do padrão HEVC divide cada quadro do vídeo em blocos quadrados chamados de *Coding Tree Units* (CTU), que tem o tamanho de 64x64 *pixels* (SULLIVAN, 2012). Cada CTU pode ainda ser dividida em blocos de 24 tamanhos diferentes desde 4x8 até 64x64 amostras para que, só então, a ME aplique os algoritmos de busca para cada um destes blocos. Estes algoritmos procuram o bloco de referência mais semelhante ao bloco sendo codificado e, então, geram o valor da diferença entre os blocos e um vetor de movimento que representa a posição do bloco encontrado com relação ao bloco atual, obtendo assim maior eficiência na representação de cada quadro do vídeo e diminuindo o espaço de armazenamento necessário.

Para obter a diferença entre dois blocos diferentes critérios de similaridade podem ser utilizados, porém a Soma das Diferenças Absolutas (SAD – *Sum of Absolute Differences*) possui uma implementação eficiente visto que o seu respectivo *hardware* não necessita de multiplicadores. Para o cálculo de SAD, inicialmente a diferença absoluta é obtida, amostra por amostra, entre dois blocos (AFONSO, 2016) e, após isso, as diferenças de todas as amostras são somadas.

Na literatura, existem vários trabalhos que propõem arquitetura de hardware para algum destes dois estágios, como AFONSO (2016) E FAN (2017), porém apenas alguns trabalhos, como PASTUSZAK (2016), propõem uma arquitetura de hardware para ambos os estágios da ME. Assim, este trabalho apresenta uma arquitetura com uma alta taxa de processamento para ambos os estágios da ME, possuindo módulos específicos para lidar com três tamanhos de blocos: 8x8, 16x16 e 32x32.

2. METODOLOGIA

Como a ME é uma etapa altamente complexa, a utilização de estratégias para reduzir esta complexidade são altamente desejáveis. Uma estratégia simples é diminuir a quantidade de tamanhos de blocos a serem testados pela ME, diminuindo assim o número de comparações necessárias para encontrar a melhor estrutura de particionamento de cada quadro. Baseado nas avaliações apresentadas por AFONSO (2016), sobre os tamanhos de bloco mais selecionados e representativos, o impacto da estratégia de utilizar apenas três tamanhos de bloco (8x8, 16x16 e 32x32) foi avaliado. Os experimentos foram realizados utilizando 24 sequências de vídeo (Classes A-F) de acordo com as Condições Comuns de Teste (CTC – *Common Test Conditions*) (BOSSEN, 2011), utilizando o software de referência do padrão HEVC (HM – *HEVC Test Model*) (HEVC, 2017). A Tabela 1 apresenta a redução no tempo de codificação e o aumento no BD-rate referentes ao uso desta estratégia. Os resultados demonstram uma diminuição de 65,7% no tempo total de codificação com um aumento médio de 4,75% no BD-rate, que representa o percentual de variação na taxa de *bits* para a mesma qualidade visual (BJONTEGAARD, 2008).

Tabela 1. Percentual de variação no BD-rate e no tempo de codificação.

Classes	Aumento no BD-Rate (%)	Diminuição do Tempo (%)
Classe A - 2560x1600	4,83	65,50
Classe B - 1920x1080	3,53	66,55
Classe C - 832x480	4,34	65,91
Classe D - 416x240	5,07	64,72
Classe E - 1280x720	5,53	65,96
Classe F- diversas	5,61	65,58
Média	4,75	65,70

De posse dos resultados de impacto na compressão ao utilizar a estratégia de redução de complexidade, uma arquitetura que implementa os dois estágios da ME foi desenvolvida. Para isso foram desenvolvidos quatro, dois e um módulos para processar blocos de tamanho 8x8, 16x16 e 32x32, respectivamente. Cada um destes módulos possui uma instância da IME e uma instância da FME, e todos os módulos operam em paralelo. A descrição dos algoritmos e do funcionamento de um módulo são apresentados na sequência do texto.

Por padrão, o software de referência do HEVC adota o *Test Zone Search* (TZS) como algoritmo para a IME. Este algoritmo é composto por quatro passos que são: a Predição, a Busca Inicial, o *Raster* e o Refinamento (LI, 2014). A Predição é responsável por definir o ponto de partida para a Busca Inicial, embora neste trabalho apenas o bloco colocalizado seja utilizado na predição.

Na Busca Inicial são selecionados os blocos mais próximos ao ponto de partida para comparação. Em seguida, a seleção expande para blocos mais distantes seguindo um formato de losango, conforme visto no esquema da Figura 1-a. A expansão termina caso o limite da área de busca seja atingido, ou caso a seleção expanda três vezes consecutivas sem obter um melhor resultado.

A etapa *Raster* é uma busca completa subamostrada, realizada apenas

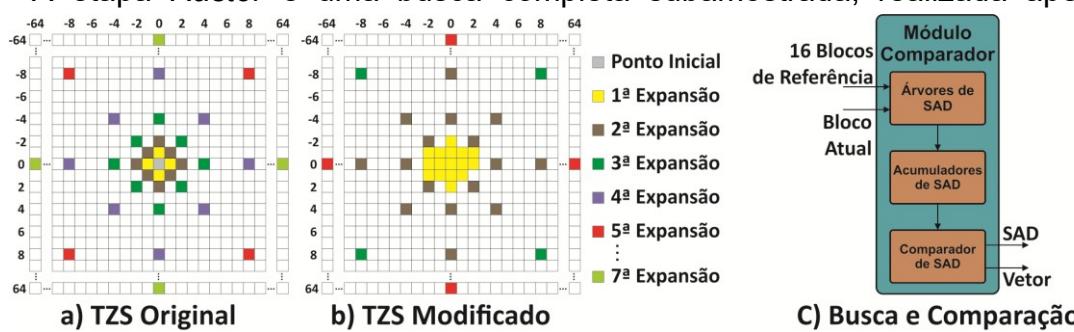


Figura 1 – Esquema de expansões do TZS (a) e (b); Módulo de comparação de blocos (c)

quando algumas condições são satisfeitas, e não foi utilizada no desenvolvimento deste trabalho. Já o Refinamento utiliza o mesmo esquema de expansões e condições de parada da Busca Inicial. Porém, o Refinamento utiliza como ponto inicial o resultado do passo anterior. Além disso, o Refinamento pode ser executado várias vezes enquanto for obtido um bloco mais similar ao bloco atual.

A parte do módulo desenvolvido para a IME implementa uma versão modificada do algoritmo TZS, que foi previamente avaliada no HM. Neste algoritmo modificado todas as expansões sempre recebem 16 blocos de referência. Para isto algumas expansões foram unificadas, conforme o esquema de expansões modificado exibido na Figura 1-b. Na arquitetura desenvolvida, um controle é responsável por lidar com o algoritmo do TZS, enquanto a parte operativa, vista na Figura 1-c, é responsável por realizar a comparação dos blocos de referência com o bloco atual utilizando 16 árvores de SAD, 16 acumuladores e um comparador de SAD. Ao final do processamento, o bloco com menor SAD e o seu respectivo vetor de movimento são obtidos.

Para a FME são utilizados alguns filtros *Finite Impulse Response* (FIR) para gerar novas amostras em posições fracionárias a partir do resultado da IME (SULLIVAN, 2012). O hardware desenvolvido para a FME possui uma unidade de interpolação (UI) que gera 48 novos blocos possíveis, além de uma unidade de busca e comparação (UBC) que realiza a comparação dos blocos gerados pela UI de forma semelhante à parte operativa da IME apresentada na Figura 1-c, porém utilizando 12 árvores de SAD e 48 acumuladores.

A arquitetura completa pode ser vista na Figura 2, onde o conjunto de módulos para todos os tamanhos de blocos utilizados pode ser visto. Uma unidade global acumula o SAD resultante de cada módulo e, após os módulos terminarem o processamento de uma CTU, ele compara estes SADs para encontrar o conjunto de blocos que resulte no menor SAD para codificar a CTU, garantindo assim uma codificação eficiente.

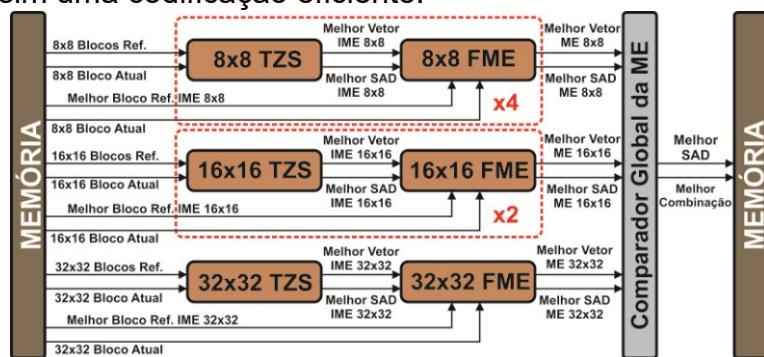


Figura 2 – Arquitetura completa desenvolvida

3. RESULTADOS E DISCUSSÃO

A arquitetura foi descrita em VHDL e a síntese ASIC foi realizada utilizando a ferramenta RTL Compiler (CADENCE, 2017). Os resultados obtidos estão detalhados na última coluna da Tabela 2.

A arquitetura proposta por AFONSO (2016) possui uma menor área e consumo do que a nossa arquitetura desenvolvida, porém ela implementa apenas a FME. A arquitetura proposta por FAN (2017) implementa apenas a IME, sendo que ele utilizou um algoritmo similar ao TZS, porém testando muito mais blocos na Busca Inicial e com apenas um pequeno refinamento, assim a taxa de processamento máxima suportada é de 30 quadros por segundo em resolução UHD 2160p (3840x2160 pixels).

Já PASTUSZAK (2016) implementou toda a ME, porém adotou várias simplificações nos algoritmos para conseguir atingir a taxa de processamento

necessária para processar vídeos com a resolução de UHD 2160p (3840x2160 *pixels*) a 30 quadros por segundo. Dentre as simplificações, ele processa apenas blocos de tamanho 8x8, e o resultado para blocos maiores são preditos a partir deste resultado. Além disso, não foi apresentado como a taxa de processamento máxima suportada pela arquitetura foi calculada, dada a imprevisibilidade do TZS. A arquitetura desenvolvida neste trabalho é a única capaz de realizar a ME para vídeos UHD 4320p (7680x4320 *pixels*) a 55 quadros por segundo (taxa adequada para processamento em tempo real).

Tabela 2. Resultados e trabalhos relacionados usando a tecnologia ASIC.

Trabalhos Relacionados	Afonso	Fan	Pastuszak	Arquitetura Desenvolvida
IME/FME	não / sim	sim / não	sim	sim
Algoritmo da IME	não	TZS Modificado	TZS Modificado	TZS Modificado
Tecnologia ASIC	Nangate 45nm	65nm	TSMC 90nm	Nangate 45nm
Área Total (gates)	148,4k	489,4k	422,7k	8371k
1080p@30fps	Frequência (MHz)	49,6	125	100
	Potência/Tensão	4,96mW/0,9V	N.D.	N.D.
2160p@30fps	Frequência (MHz)	N.D.	500	400
	Potência/Tensão	N.D.	128,5mW/N.D.	293mW/N.D.
4320p@30fps	Frequência (MHz)	não	não	não
	Potência/Tensão	não	não	896,52mW/0,95V

4. CONCLUSÕES

Este trabalho apresentou uma arquitetura para a etapa de ME segundo o padrão HEVC. Os resultados mostraram que a arquitetura desenvolvida pode atingir frequências de operação de até 1,14 GHz e processar vídeos com resoluções UHD 4320p (7680x4320 *pixels*) a 55 quadros por segundo com uma dissipação de 896,52 mW, sendo a única capaz de processar vídeos UHD 4320p em tempo real.

5. REFERÊNCIAS BIBLIOGRÁFICAS

- AFONSO, V. et al. "Hardware Implementation for the HEVC Fractional Motion Estimation Targeting Real-Time and Low-Energy". **Journal of Integrated Circuits and Systems**, v.11, n.2, p.106–120, 2016.
- BJONTEGAARD, G. **Improvements of the BD-PSNR model**, VCEG-AI11, Jul, 2008.
- BOSSEN, F. "Common Test Conditions and Software Reference Configurations", **JCTVC-L1100**, Geneva, Jan 2013.
- CADENCE. **Cadence RTL Compiler**. Acesso em Fevereiro 2017. Disponível em: <<https://www.cadence.co.jp/RTL/>>.
- FAN, Y. et al. "A Hardware-Oriented IME Algorithm for HEVC and its Hardware Implementation". **IEEE Transactions on Circuits and Systems for Video Technology**, n. 99, 2017.
- HEVC. **HEVC Reference Software**. Acesso em: Fevereiro 2017. Disponível em: <<http://hevc.hhi.fraunhofer.de>>.
- ITU-T. "H.265 High efficiency video coding". **Recommendation ITU-T**, 2013.
- LI, X. et al. "Fast motion estimation methods for HEVC". **IEEE International Symposium on Broadband Multimedia Systems and Broadcasting**, p. 1–4. 2014
- OHM, J. R. et al. "Comparison of the Coding Efficiency of Video Coding Standards x2014; Including High Efficiency Video Coding (HEVC)". **IEEE Transactions on Circuits and Systems for Video Technology**, v.22, n.12, Dez 2012.
- PASTUSZAK, G.; TROCHIMIUK, M. "Algorithm and architecture design of the motion estimation for the H. 265/HEVC 4K-UHD encoder". **Journal of Real-Time Image Process.**, v.12, n.2, p.517–529, Ago 2016.
- SULLIVAN, G. J. et al. "Overview of the High Efficiency Video Coding (HEVC) Standard". **IEEE Transactions on Circuits and Systems for Video Technology**, v.22, n.12, p.1649–1668, Dez 2012.