

SIMPLIFICAÇÃO ALGORÍTMICA NOS FILTROS DE INTERPOLAÇÃO DO CODIFICADOR AV1 VISANDO A IMPLEMENTAÇÃO EM HARDWARE

WILLIAM KOLODZIEJSKI¹; ROBSON ANDRÉ DOMANSKI²; LUCIANO AGOSTINI³

¹ UFPel – wkolodziejski@inf.ufpel.edu.br

² UFPel – radomanski@inf.ufpel.edu.br

³ UFPel – agostini@inf.ufpel.edu.br

1. INTRODUÇÃO

O número de dispositivos capazes de reproduzir vídeos digitais bem como a demanda por resoluções elevadas – como o UHD (*Ultra High Definition*) 4K (3840x2160) e 8K (7680x4320) – são cada vez maiores. Statista (2020) estima que até 2022, os vídeos digitais serão responsáveis pelo tráfego de cerca de 77,49 EiB/mês (1 EiB igual a 2^{60} bytes), sendo esse valor correspondente a cerca de 82% de todo o tráfego global da Internet (CISCO, 2018). Levando em conta esse cenário, a criação de novos padrões de codificação torna-se necessário para diminuir o número de bytes usados para representar os vídeos. O codificador AV1 (AOM Video 1) da *Alliance for Open Media* é um exemplo.

O AV1 foi lançado em 2018 como um novo codificador de vídeo, criado por uma aliança de empresas gigantes do ramo (Google, Mozilla, Intel, etc.) com o objetivo de oferecer grandes taxas de compressão, com altíssima qualidade de imagem e que fosse livre de *royalties*. O novo padrão é capaz de entregar cerca de 17~25% (GOIS, 2017); (GUO, 2018) mais compressão do que seu antecessor VP9, porém, para obter tal feito, a complexidade computacional aumentou em mais de 100 vezes (GROIS, 2017), o que leva a um aumento no consumo de energia, no tempo necessário de codificação e na largura de banda de memória necessária.

Esse aumento no custo computacional deve-se ao uso de novas ferramentas de codificação mais complexas e de melhorias em outras já existentes, como é o caso da Estimação de Movimento Fracionária (FME), que foi herdada de seu antecessor VP9 e foi expandida (CHEN, 2018). Essa ferramenta realiza cálculos de interpolação a partir de amostras inteiras, a fim de obter sub pixels fracionários para representar movimentos mais suaves, proporcionando um maior refinamento na etapa de Estimação de Movimento Inteira (IME). Para realizar a interpolação, o AV1 conta com 90 filtros, sendo esses divididos em 6 famílias: REGULAR 4 e 8 TAPs, SMOOTH 4 e 8 TAPs, SHARP 8 TAPs e BILINEAR (2 TAPs). Cada família contém 15 filtros, conforme especificação da AOMedia (2019).

O desenvolvimento de Circuitos Embarcados para codificação de vídeo tem se tornado cada vez mais necessário, principalmente em dispositivos móveis, onde a redução no consumo de energia e na complexidade do processamento são questões críticas. Para atingir esses objetivos, uma técnica bastante utilizada é a aplicação de computação aproximada (BARUA, 2019). Nessa técnica são propostas simplificações no algoritmo ou no projeto do circuito integrado, para reduzir o custo computacional e o consumo de energia, ao custo de perda na precisão do cálculo. Tendo em vista a capacidade que a codificação de vídeos tem em ser resiliente a erros, a proposta de um circuito que aplica computação aproximada na FME torna-se muito conveniente, como realizado em trabalhos para outros codificadores, como o de PENNY (2018). Neste resumo, apresentaremos os resultados obtidos em experimentos com algumas propostas de implementação de filtros aproximados, para que, posteriormente, sejam implementados em hardware.

2. METODOLOGIA

Após estudos sobre a FME, através do código fonte do AV1 e da documentação disponibilizada (AOMedia, 2019), foram elaborados filtros alternativos, que deverão substituir os originais em uma implementação em hardware futura, com o intuito de oferecer uma redução significativa no custo computacional da ferramenta. Tendo em vista que o hardware a ser proposto utilizará computação aproximada, os filtros de interpolação foram modificados visando simplificar a implementação desse circuito, de forma que seus valores (TAPs) são definidos por deslocamentos e/ou somas e deslocamentos. Para compreender esse processo, é necessário entender como funciona a interpolação e as operações aritméticas em hardware.

A interpolação assemelha-se a um cálculo de média ponderada, onde as amostras (pixels) serão multiplicadas por pesos (TAPs), acumuladas e, no fim, será realizada a divisão inteira dessa acumulação pela soma dos pesos (no caso do AV1, 128). Esse valor será chamado sub pixel e representará um pixel que não existe originalmente no vídeo, mas seu valor fornece uma estimativa de movimento mais precisa do que outro pixel existente.

As operações aritméticas em hardware são implementadas usando unidades chamadas somadores, subtratores, multiplicadores e divisores, sendo que essas duas últimas utilizam vários somadores e subtratores para realizar suas operações. Nos circuitos digitais, os multiplicadores e divisores estão entre os elementos que mais consomem energia e que são mais lentos. Entretanto, é possível realizar multiplicações e divisões usando deslocamentos, que utilizam muito menos energia e tempo. Porém apenas multiplicações e divisões por potências de dois são possíveis (devido ao funcionamento do sistema binário). Para fins de comparação, dependendo da implementação, um multiplicador tem seu tempo de computação diretamente proporcional ao tamanho do dado de entrada, sendo geralmente responsável pelo maior atraso em um circuito (AHMED, 2015), enquanto um deslocador tem tempo constante e atraso baixíssimo, pois trata-se apenas de fios.

Tendo em vista o funcionamento dos deslocadores, os valores dos TAPs dos filtros foram modificados de forma a utilizarem as potências de dois mais próximas aos valores originais. Tudo isso para ser possível projetar um hardware capaz de realizar as interpolações com resultados próximos o suficiente dos originais, mas com menor custo energético e computacional.

Foram propostos então três aproximações nos filtros, todos usando deslocamentos: (i) ALT1 (6 famílias de 15 filtros) que usa apenas um deslocamento para cada TAP. Por exemplo, o valor 75 seria representado por $2^6 = 64$; (ii) ALT2 (1 família de 15 filtros) que usa apenas um deslocamento para cada TAP, mas não implementa as demais famílias; e (iii) ALT3 (6 famílias de 15 filtros) com a adição de dois somadores intermediários, buscando um menor erro através da soma ou subtração de até três deslocamentos em alguns TAPs. Por exemplo, o número 75 pode ser aproximado por $2^6 + 2^3 + 2^1 = 74$, com um erro muito menor do que usando apenas um deslocamento.

Após as alterações realizadas nos valores dos TAPs dos filtros, foram conduzidos diversos testes, onde, utilizando sequências de vídeo padronizadas para pesquisas, foram codificados a 60 quadros de cada vídeo em resoluções de Full HD e 4K, com parâmetros de quantização (GERSHO, 1997) definidos em 20, 32, 43 e 55 para cada filtro proposto, além do filtro original.

As abordagens de aproximação foram avaliadas usando o BD-rate, que é a métrica utilizada para medir a eficiência de codificação, relacionando qualidade do vídeo (PSNR) com a quantidade de bits necessários para sua representação.

Quanto menor o BD-rate, menor é o número de bits necessários para representar o vídeo com uma mesma qualidade, se comparado com a implementação original. Assim, quanto menor o valor de BD-rate, melhor. Cada sequência analisada gerou três valores de BD-rate, sendo um valor por abordagem de aproximação e, com isso, foi possível obter a média geral da perda de eficiência que cada filtro inseriu por conta das aproximações, em comparação com o filtro original.

3. RESULTADOS E DISCUSSÃO

A Tabela 1 apresenta os resultados de BD-rate obtidos na avaliação das três abordagens de aproximação para os filtros de interpolação da FME do AV1, considerando as sequências de vídeo avaliadas. A média também é apresentada.

Comparação BD-rate (%)				
Sequência	Resolução	ALT1	ALT2	ALT3
Arena Of Valor	Full HD	3,28	4,71	0,31
Market Place		5,27	9,78	1,42
Square And Timelapse		7,32	11,46	1,33
Tunnel Flag		10,9	16,85	2,84
Foreman	4K	1,54	2,49	0,27
Coastguard		-0,17	-0,29	-0,17
Cactus		1,01	1,90	1,70
Média	Full HD	6,69	10,70	1,47
	4K	0,79	1,36	0,60

Tabela 1 - Quanto menor, melhor

Durante as avaliações, as sequências em Full HD foram as que apresentaram maiores taxas de perda de qualidade, enquanto as em 4K não apresentaram perdas tão expressivas. Isso deve-se principalmente à quantidade de pixels presentes nas diferentes resoluções, o que permite que o 4K tolere mais erros de aproximação sem comprometer a qualidade de forma tão agressiva.

Os resultados seguem a tendência esperada, com as maiores perdas de eficiência na abordagem ALT2, seguida pela abordagem ALT1 e pela abordagem ALT3, especialmente para vídeos Full HD, que ficaram mais elevadas do que o esperado usando as abordagens ALT1 e ALT2. Por outro lado, o filtro ALT3 é o que apresenta melhores resultados em relação à perda de qualidade – já que utiliza aproximação de forma menos agressiva – sendo esse o principal candidato a ser implementado em hardware, que será a próxima etapa do trabalho. Outra possibilidade a ser explorada é gerar uma FME configurável, ou seja, a partir de parâmetros externos, como a carga da bateria, por exemplo, é definido qual estratégia de aproximação será usada. Assim pode-se optar por uma estratégia mais agressiva de aproximação, para atingir maiores ganhos no consumo de energia, mesmo que isso traga uma maior perda na eficiência de codificação.

4. CONCLUSÕES

Tendo em vista que esse é o primeiro trabalho da literatura relacionado ao uso de computação aproximada nos filtros de interpolação da FME do AV1, os autores esperam fornecer uma boa alternativa de implementação de hardware para dispositivos que necessitem circuitos com baixo consumo de energia e tolerem alguma perda de eficiência de codificação, além de contribuir para futuros trabalhos a serem propostos na literatura.

5. REFERÊNCIAS BIBLIOGRÁFICAS

Statista. **Global mobile data traffic from 2017 to 2022**. Acessado em 17 set. 2020. Online. Disponível em: <https://www.statista.com/statistics/271405/global-mobile-data-traffic-forecast/>

Cisco. **Cisco Predicts More IP Traffic in the Next Five Years Than in the History of the Internet**. Acessado em 18 set. 2020. Online. Disponível em: <https://newsroom.cisco.com/press-release-content?type=webcontent&articleId=1955935>

GROIS, D.; NGUYEN, T.; MARPE, D. Performance Comparison of AV1, JEM, VP9, and HEVC Encoders. **Applications of Digital Image Processing XL**, 2017.

GUO, L.; COCK, J.D.; AARON, A. Compression Performance Comparison of x264, x265, libvpx and aomenc for On-Demand Adaptive Streaming Applications. **Picture Coding Symposium (PCS)**, San Francisco, 2018.

AOMedia. **Av1 bitstream e decoding process specification**. Alliance for open media. Acessado em 22 set. 2020. Online. Disponível em: <https://aomediacodec.github.io/av1-spec/av1-spec.pdf>

PENNY, W. *et al.* Power-Efficient and Memory-Aware Approximate Hardware Design for HEVC FME Interpolator. **25th IEEE International Conference on Electronics, Circuits and Systems (ICECS)**, Bordeaux, 2018.

AHMED, A. *et al.* Fixed Point and Floating Point High Speed Hardware Multipliers - A comparison of Bit Serial and Wallace Tree Multipliers Using Booth Recoding. **International Journal of Engineering & Technology IJET-IJENS**, v.15 n.1, 2015.

CHEN, Y. *et al.* An Overview of Core Coding Tools in the AV1 Video Codec. **Picture Coding Symposium (PCS)**, San Francisco, 2018.

BARUA, H.B.; MONDAL, K.C. Approximate Computing: A Survey of Recent Trends - Bringing Greenness to Computing and Communication. **The Journal of the Institution of Engineers: Electronics and Telecommunication Division**, India, p.619–626, 2019.

GERSHO, A. Quantization. **IEEE Communications Society Magazine**, v.15, n.5, 1977.