AVALIAÇÃO DA ESCALABILIDADE DE PARALELISMO DO PADRÃO DE CODIFICAÇÃO AV1 ATRAVÉS DA UTILIZAÇÃO DE TILES

BERNARDO BELING; IAGO STORCH; DANIEL PALOMINO

Universidade Federal de Pelotas – Video Technology Research Group (ViTech)
{berbeling, icstorch ,dpalomino}@inf.ufpel.edu.br

1. INTRODUÇÃO

Segundo a CISCO, há uma previsão de que até 2022 82% da banda utilizada na internet seja de vídeos e streaming no geral (CISCO, 2019). Netflix, YouTube, HBOGO e Facebook estão entre os serviços de streaming mais populares atualmente, e embora as resoluções HD (1280×720 pixels) e FullHD (1920×1080 pixels) sejam as mais utilizadas, vídeos em resolução 4K (3840×2160 pixels) têm ganhado espaço. Contudo, um vídeo digital em seu formato original demanda uma quantidade de dados muito grande para ser transmitido: um vídeo HD de 2 minutos, apresentado a uma taxa de 30 quadros por segundo e possuindo 3 canais de cores (amostras) por pixel, onde cada canal de cor é representado por 1 byte, possui tamanho 9,95 GB (1280×720×120×30×3). Com isso, surge a necessidade da compressão de vídeo que visa reduzir o volume de dados utilizado para representar o vídeo com o mínimo de perdas de qualidade possível.

Considerando que um vídeo é uma sequência de imagens (denominados quadros) exibidas a pelo menos 24 quadros por segundo de modo a formar uma continuidade visual, existe muita similaridade entre quadros sucessivos. Além disso, dentro de um mesmo quadro costuma existir regiões muito homogêneas. Sendo assim, a compressão é feita através da exploração de redundâncias espaciais (dentro do mesmo quadro) e temporais (entre quadros vizinhos).

Para realizar tal tarefa, existem diversos padrões de codificação de vídeo, e entre eles é possível destacar o *H.265 High Efficiency Video Coding* (HEVC), proposto em 2013, que é capaz de manter a mesma qualidade que o seu antecessor H.264 utilizando 35% menos bytes (CIANET, 2015). Em contrapartida, a utilização do HEVC não atingiu uma grande crescente devido as altas taxas de royalties cobradas pelos detentores do HEVC, fazendo com que muitas empresas não adotem o uso do padrão. Devido a isso, em 2015 a *Alliance for Open Media* (AOMedia), formada por grandes empresas da área de tecnologia como Google, Amazon, Netflix, Microsoft, entre outras, se uniram visando a criação de um padrão de codificação livre de royalties, denominado AV1.

No cenário atual, por se tratar de um padrão emergente, o AV1 ainda possui uma performance inferior ao HEVC, mas promete ser 30% mais eficiente que o HEVC (IBC, 2018) quando seu desenvolvimento estiver concluído. Todavia, assim como acontece em outros padrões de codificação, uma codificação AV1 possui alta demanda computacional e gera uma carga de trabalho elevada nas unidades de processamento do sistema utilizado para codificação.

Para distribuir a alta carga computacional entre as unidades de processamento gerada pela codificação de vídeo, o padrão AV1 propõe o uso da ferramenta de paralelismo Tiles. Tiles são regiões retangulares no quadro delimitadas por limites horizontais e verticais, sendo que essas divisões podem ser uniformes (mesma altura e largura para todas divisões) ou não. Cada uma dessas regiões é codificada de forma independente, pois não existem dependências de dados entre os Tiles. O tamanho de um Tile é medido através

da unidade de codificação básica do AV1, os Superblocks (SBs), que possuem tamanho 128x128 pixels. Sendo assim, Tiles tem dimensões múltiplas de 128.

O uso de Tiles uniformes durante o vídeo, embora o divida em regiões de mesmas dimensões, não divide igualmente a carga de trabalho entre eles (STORCH, 2016). Isso se dá, pois essa carga de trabalho é determinada pela quantidade de detalhes da região do vídeo. Desse modo, alguns Tiles podem englobar regiões mais homogêneas e estáticas do que outros, gerando cargas de trabalho desbalanceadas entre Tiles.

Em (PAPADOPOULOS, 2018) os autores realizaram testes experimentais para avaliar o impacto da utilização de Tiles no AV1, relacionando a quantidade de núcleos de processamento utilizados com o nível de paralelismo extraído. Para isso, foram realizados três tipos de particionamentos com 4, 6 e 8 núcleos/Tiles. Todavia, essa quantidade de particionamentos é baixa e não é suficiente para demonstrar de fato o comportamento dos Tiles.

Sendo assim, neste trabalho é medida a escalabilidade do paralelismo no codificador de referência do padrão AV1 utilizando desde 1 até 12 núcleos de processamento, além de avaliar a codificação com e sem a utilização de Tiles. Por fim, é comparada a aceleração da codificação aderindo ao uso de Tiles em relação a uma codificação sem Tiles.

2. METODOLOGIA

Este trabalho avalia a escalabilidade do codificador de referência do padrão AV1 de duas formas: realizando codificações sem o uso de Tiles, para medir a escalabilidade própria do codificador conforme a quantidade de núcleos de processamento utilizados, e realizando codificações utilizando Tiles uniformes para medir a eficiência do uso de Tiles. Essas avaliações foram feitas em dois vídeos de resolução 1920×1080 e dois vídeos de resolução 2560×1600.

Para a realização das avaliações, foi utilizado o software de referência disponibilizado pela AOMedia (AOMedia, 2019) que possui as ferramentas de compressão do padrão AV1. A importância da utilização desse software está na possibilidade de comparação com trabalhos existentes na literatura do AV1. Todas as codificações foram feitas num servidor equipado com um processador Intel Xeon E5-2640 2.20GHz com 12 núcleos físicos e 64GB de memória RAM.

Com o software de referência devidamente instalado, as avaliações foram executadas no terminal do servidor, e, sendo necessário efetuar diversas codificações com múltiplos parâmetros de configuração específicos a cada uma, foi desenvolvido um *script* com a linguagem de programação *python* visando automatizar as execuções. Dentre os parâmetros utilizados na codificação, foram utilizados desde 1 até 12 núcleos tanto com Tiles desabilitados quanto habilitados, o parâmetro de quantização (QP) utilizado foi de 39, e as codificações ocorreram nos 20 primeiros quadros de cada vídeo.

Nas codificações utilizando Tiles, a quantidade dos mesmos é igual a quantidade de núcleos utilizados. Além disso, como o tamanho dos Tiles é medido por unidade de SBs (128×128), as dimensões dos Tiles em SBs também devem ser informadas pelo terminal via parâmetro. Na Figura 1 é apresentada a representação de um video de resolução 1920×1080 sendo dividido em 3 unidades de Tiles verticais e 2 unidades de Tiles horizontais, totalizando 6 Tiles. Como o vídeo possui 15 SBs de largura (1920÷128=15), uma divisão uniforme faz com que cada Tile tenha 5 SBs de largura. Por outro lado, o vídeo possui 9 SBs de altura (1080÷128=8,4 arredondado para cima), então é necessário que alguns Tiles tenham 5 SBs de altura, enquanto que outros tenham 4 SBs.

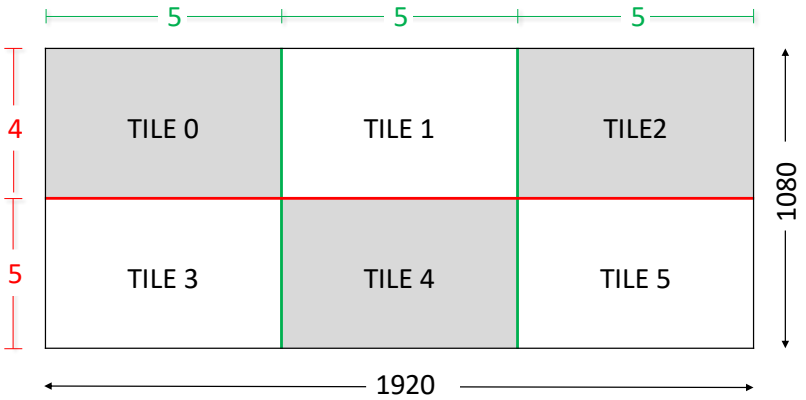

Figura 1. Divisão de 6 Tiles em um vídeo FullHD

Assim que as codificações foram concluídas, foi possível extrair os resultados de tempo de execução individual das codificações através de arquivos de texto gerados ao final de cada execução. Com isso, foi realizada uma avaliação de tempo de execução por núcleos/Tiles utilizados nas duas avaliações, com o objetivo de compreender a escalabilidade do codificador nos diferentes cenários (com e sem Tiles e em resoluções distintas).

De modo a avaliar a aceleração obtida, foi elaborado um gráfico onde o tempo de execução da codificação com 1 núcleo é a referência aos tempos das execuções com mais núcleos. Dessa forma, o tempo de execução da referênia foi dividido pelo tempo de execução de todas outras codificações com mesma resolução e mesmo vídeo, de modo a obter a aceleração em relação a codificação de referência. Por exemplo: se a codificação utilizando 1 Tile demandou 60 segundos de processamento e a codificação com 2 Tiles demandou 40 segundos de processamento, então a segunda codificação obteve uma aceleração de 1,5 em relação a codificação de referência (60÷40=1,5).

3. RESULTADOS E DISCUSSÃO

Após os passos aprensentados na metodologia serem concluídos, foram obtidos os resultados de escalabilidade de ambas resoluções. Na Figura 2, os valores aprensentados são a média de todas codificações, englobando ambas resoluções. Porém, de forma individual, os vídeos de resolução mais alta (2560×1600) apresentaram uma aceleração mais elevada em ambas avaliações. Além disso, vale lembrar que nas codificações com Tiles o número de Tiles é igual ao número de núcleos utilizados.

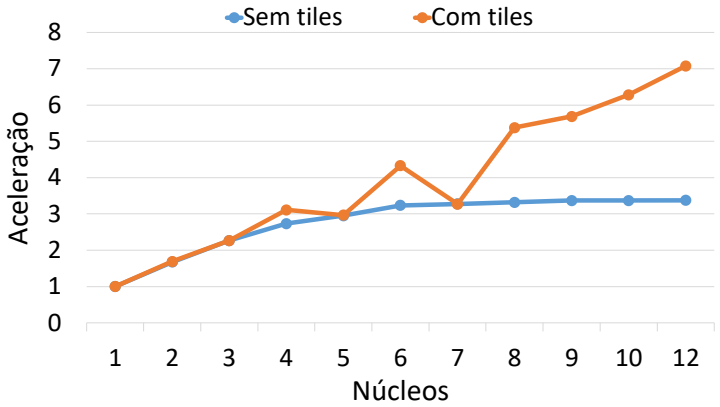

Figura 2. Escalabilidade da aceleração por número de núcleos

A partir do gráfico, percebe-se que de fato o codificador AV1 possui paralelismo próprio mesmo sem a utilização de Tiles, e que a aceleração tende a aumentar conforme mais núcleos/Tiles são utilizados. No entanto, nas codificações sem Tiles a aceleração tende a saturar a partir da utilização de 6 núcleos, enquanto a utilização de Tiles não apresentou um ponto de saturação.

Também é possível observar que nas codificações com Tiles há uma queda de aceleração na utilização 5 e 7 núcleos, obtendo um desempenho próximo à codificação sem o uso de Tiles. Isso acontece devido a falta de uniformidade nas dimensões dos Tiles com esse número de núcleos, pois 5 e 7 são números primos, fazendo com que só existam Tiles verticais ou horizontais

Apesar da aceleração nas codificações utilizando Tiles apresentarem um comportamento similar a uma reta, é notável que essa não é propocional a quantidade de Tiles utilizados, pois as codificações utilizando 12 Tiles foram em média 7 vezes mais rápida em relação a codificação de referência, ao contrário do ideal que seria 12 vezes mais rápida.

4. CONCLUSÕES

Com os resultados apresentados anteriormente, pode-se concluir que há uma melhor distribuição na carga de trabalho entre as unidades de processamento quando é utilizada a ferramenta de Tiles, que por sua vez, gerou uma aceleração nas codificações. Todavia, essa aceleração não ocorre de forma linear com a quantidade de núcleos/Tiles utilizados por codificação, o que demonstra uma margem passível à explorações para melhorias.

Em (STORCH, 2016), é apresentanda uma técnica de particionamento de Tiles para o HEVC que os distribue dinamicamente por quadro de acordo com a carga de trabalho gerada por Tiles anteriores. Sendo assim, como trabalho futuro pretende-se adaptar esse algoritmo ao padrão AV1 para melhorar a distribuição de carga de trabalho entre os Tiles deste padrão.

5. REFERÊNCIAS BIBLIOGRÁFICAS

CISCO, **Visual Networking Index: Forecast and Trends 2017-2022 Whitepaper.** Acessado em: 3 set. 2019. Online. Disponível em: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html

Cianet. **Saiba o que é o novo formato H.265**. Acessado em: 4 set. 2019. Online. Disponível em: https://www.cianet.com.br/blog/inovacao-e-tendencias/saiba-o-que-e-o-novo-formato-h-265/

IBC. **Codec Wars: The battle between HEVC and AV1**. Acessado em 4 set. 2019. Online. Disponível em: https://www.ibc.org/publish/codec-wars-the-battle-between-hevc-and-av1/2710.article

STORCH, I. SPEEDUP-AWARE HISTORY-BASED TILING ALGORITHM FOR THE HEVC STANDARD. **2016 IEEE Internacional Conference on Image Processing (ICIP).**

PAPADOPOULOS, P. On the Evaluation of Coarse Grained Parallelism in AV1 Video Coding. **2018 13th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP).**

AOMedia. Repositório git da Alliance for Open Media. Disponível em: https://aomedia.googlesource.com/aom/