

1.03.04 - Ciência da Computação / Sistemas de Computação  
**TÉCNICA DE PARTICIONAMENTO DE TILES DINÂMICA VISANDO O AUMENTO DE SPEEDUP DO  
PROCESSO DE CODIFICAÇÃO DE VÍDEOS SEGUNDO O PADRÃO HEVC**

Iago C. Storch<sup>1</sup>, Daniel Palomino<sup>2</sup>, Luciano Agostini<sup>3</sup>

1. Estudante de Engenharia de Computação e bolsista PIBIC da UFPel

2. Doutor em Computação e Professor do PPG em Computação da UFPel

3. Doutor em Computação e Professor do PPG em Computação da UFPel / Orientador

**Resumo:**

O processo de codificação de vídeo exige grande poder computacional em função da enorme quantidade de dados que precisam ser processados e da complexidade dos algoritmos utilizados. O padrão de codificação HEVC introduziu uma nova ferramenta de exploração de paralelismo para reduzir o tempo de processamento denominada *Tiles*. Os *Tiles* particionam cada quadro do vídeo em regiões que são codificadas de forma independente. No entanto, o particionamento padrão de *Tiles* não considera que diferentes conteúdos de vídeo apresentam diferentes demandas computacionais, o que faz com que a carga de trabalho não fique bem distribuída entre os *Tiles* e o paralelismo não seja explorado da melhor maneira possível. Este trabalho propõe uma técnica de particionamento de *Tiles* que se adapta ao conteúdo do vídeo e que é capaz de melhorar o desempenho do codificador HEVC. Resultados mostram que a técnica proposta melhora o paralelismo do codificador HEVC, sem introduzir perdas na eficiência de codificação.

**Palavras-chave:** paralelismo; balanceamento de carga de trabalho; ganho de desempenho

**Apoio financeiro:** CNPq, CAPES e FAPERGS.

**Introdução:**

Aplicações que manipulam vídeos digitais estão cada vez mais comuns em um crescente número de aplicações e isso só é possível devido à compressão destes vídeos. A compressão é atingida através do uso de diversos algoritmos definidos nos padrões de codificação de vídeo. O padrão HEVC [4] é o mais novo padrão de compressão de vídeos e usa algoritmos melhorados herdados de padrões anteriores e incorpora algoritmos inovadores. O HEVC atinge uma taxa de compressão entre 36% e 70% maior do que o padrão H.264 [8], que dominava o mercado até então, sem diminuir a qualidade dos vídeos [8]. O aumento na compressão, entretanto, também causou um aumento significativo no custo computacional necessário para comprimir os vídeos [5].

Buscando reduzir o impacto deste aumento de custo computacional, o HEVC incorporou uma ferramenta de processamento paralelo chamada de *Tiles*. *Tiles* consistem em arranjos retangulares do quadro do vídeo e são compostos por um número inteiro de CTUs (blocos básicos em que os quadros são divididos), de forma que cada *Tile* pode ser processado de forma independente [6].

Ao usar *Tiles* é possível que estes dividam o quadro em retângulos de tamanho uniforme ou não-uniforme, mas uma vez que estes tamanhos são definidos, eles serão os mesmos para todos os quadros [6]. Este método de funcionamento, no entanto, pode não explorar o paralelismo da melhor maneira possível. Uma vez que as técnicas de codificação de vídeo exploram redundâncias, áreas com muito movimento e textura demandam um maior esforço computacional do que áreas estáticas e homogêneas [9]. Como é necessário que todos os *Tiles* de um quadro tenham terminado o seu processamento para que o próximo quadro seja codificado, uma distribuição de carga de trabalho desequilibrada faz com que os *Tiles* menos custosos fiquem ociosos enquanto que os mais custosos ainda estejam realizando seu processamento [9].

Existem trabalhos propondo métodos para melhorar a distribuição de carga, como em [1], mas o modelo desenvolvido pelos autores é baseado em informações estáticas e pode não ser adequado para todos os tipos de conteúdo, além dos resultados serem limitados e possivelmente não conclusivos.

Este trabalho apresenta uma nova técnica de particionamento dinâmico de *Tiles*, que é capaz de adaptar o particionamento às características do vídeo e, com isso, equilibrar a distribuição de carga de trabalho entre os *Tiles*, melhorando o paralelismo proveniente dos mesmos.

**Metodologia:**

O primeiro passo deste trabalho foi modificar o *software* de referência do HEVC (HM-16.0) [7] para extrair o tempo de codificação por CTU. Em seguida, um conjunto de vídeos foi codificado e os tempos de codificação por CTU foram analisados. Com base nestes dados, constatou-se que quadros vizinhos apresentam distribuição de carga de trabalho muito semelhante, devido à correlação temporal. A Figura 1 (a) e (b) representa a distribuição de carga de trabalho para os quadros 3 e 4 de um dos vídeos analisados, onde cada quadrado colorido representa uma CTU e a cor está normalizada com a carga de trabalho da CTU mais custosa.

A Figura 1 mostra que existem concentrações de CTUs mais custosas em algumas regiões do quadro, e a distribuição de carga de trabalho entre os quadros vizinhos é muito semelhante. Considerando essas características, caso um mesmo particionamento fosse aplicado em ambos os quadros, como representado pelas linhas tracejadas (neste caso, um particionamento 2x2 uniforme), a distribuição de carga de trabalho entre os *Tiles* dos dois quadros também seria muito próxima.

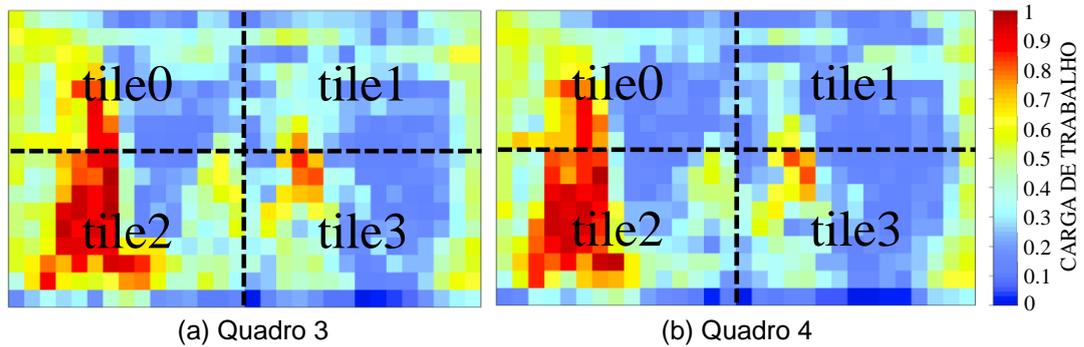


Figura 1. Distribuição de carga de trabalho do vídeo *BasketballDrive* com particionamento de *Tiles* 2x2

Considerando essa observação, foi desenvolvida uma técnica para o particionamento dinâmico de *Tiles*, baseado em histórico, para melhorar a distribuição da carga de trabalho entre os *Tiles*. Para implementar essa técnica, o HM-16.0 foi modificado para permitir que as divisões pudessem ser alteradas para cada quadro, algo que não é possível no *software* padrão. Os passos da técnica desenvolvida estão descritos na Figura 2.

Durante a codificação de um quadro, o tempo de codificação de cada CTU é armazenado em uma estrutura de matriz ( $M_{ij}$ ) que representa a distribuição de carga de trabalho. Esta informação, junto com o número de *Tiles* verticais ( $V$ ) e horizontais ( $H$ ), é utilizada como entrada para o algoritmo. O algoritmo define, então, o número de divisões verticais ( $Vd$ ) com base no número de *Tiles* verticais ( $V$ ) e calcula a carga de trabalho *alvo* para estes *Tiles*, desconsiderando os *Tiles* horizontais.

Em seguida, as linhas do histórico são avaliadas para determinar qual seria a carga de trabalho se a linha em questão fosse escolhida como divisão, considerando *linha* e *largEmCTU* (largura medida em CTUs) como limites superiores na Eq. (1). Depois de definir todas as divisões verticais, um processo semelhante é aplicado às colunas do histórico, para definir as divisões horizontais, desconsiderando as divisões verticais já definidas. Uma vez que as divisões verticais e horizontais tenham sido definidas, elas são combinadas para realizar a codificação.

| Entrada: histórico de distribuição de carga de trabalho $M_{ij}$ , número de <i>Tiles</i> verticais $V$ , número de <i>Tiles</i> horizontais $H$ |   |
|--|---|
| 1: $Vd \leftarrow V - 1$   | 8: $Hd \leftarrow H - 1$                                  |
| 2: $alvo \leftarrow cargaTotal/V$  | 9: $alvo \leftarrow cargaTotal/H$                         |
| 3: <b>para cada</b> $v$ em $Vd$ <b>faça:</b>   | 10: <b>para cada</b> $h$ em $Hd$ <b>faça:</b>             |
| 4: <b>para cada</b> linha em $M_{ij}$ <b>faça:</b>   | 11: <b>para cada</b> coluna em $M_{ij}$ <b>faça:</b>      |
| 5: $atual \leftarrow$ soma da carga até a linha  | 12: $atual \leftarrow$ soma da carga até a coluna         |
| 6: <b>se</b> $atual \approx alvo \times v$ <b>então:</b>   | 13: <b>se</b> $atual \approx alvo \times h$ <b>então:</b> |
| 7:       seleciona linha como $v$ -ésima divisão   | 14:       seleciona coluna como $h$ -ésima divisão        |

Figura 2. Pseudocódigo da técnica desenvolvida

$$atual = \sum_{i=0}^{linha\ ou\ coluna\ (larg\ ou\ alt)EmCTU} \sum_{j=0} Hist\u00f3rico_{ij} \quad (1)$$

A Figura 3 demonstra a execução do algoritmo com uso de *Tiles* 2x2. A Figura 3 (a) representa a seleção da divisão vertical, onde as linhas que foram avaliadas estão marcadas pelas linhas tracejadas enquanto que a linha selecionada está marcada com uma linha sólida. A Figura 3 (b) representa a seleção da divisão horizontal, e a Figura 3 (c) apresenta a união das duas seleções independentes.

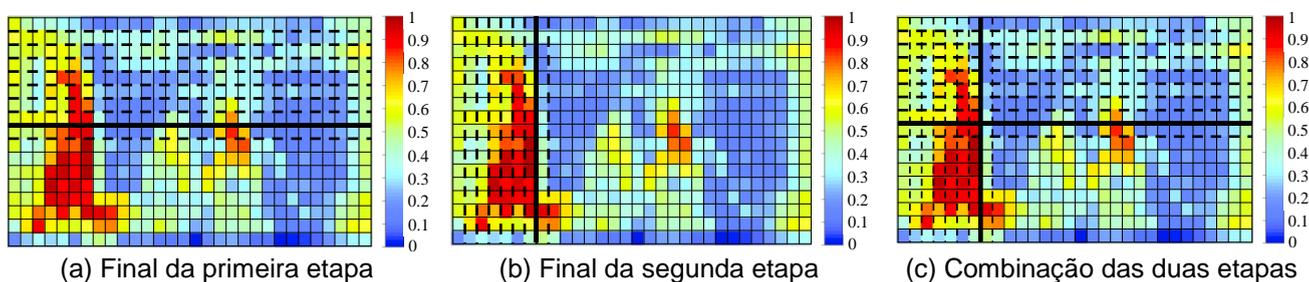


Figura 3. Etapas do algoritmo de particionamento desenvolvido

## Resultados e Discussão:

Após implementar a técnica desenvolvida no HM-16.0, um conjunto de vídeos definidos nas Condições Comuns de Teste [3] foram codificados e o desempenho do codificador modificado foi comparado ao do codificador convencional utilizando o particionamento de *Tiles* uniforme. Para avaliar o desempenho da técnica em diferentes cenários, foram empregados os particionamentos 2x2, 3x3, 4x4 e 5x5. Por fim, os resultados da técnica foram comparados com trabalhos da literatura.

O desempenho da técnica desenvolvida foi avaliado considerando dois aspectos: paralelismo e eficiência de codificação. O paralelismo foi avaliado utilizando o *speedup* e a média de tempo economizado (*Average Time Saving – ATS*). O *speedup* mede o quão mais rápida é a execução de uma tarefa utilizando processamento paralelo quando comparada com a mesma tarefa executando de forma sequencial, sendo que o *speedup* máximo é igual ao número de *Tiles* utilizados. O *ATS* mede quanto tempo foi economizado utilizando-se uma dada técnica ao invés de outra. Finalmente, a eficiência de codificação é avaliada utilizando-se o Bjøntergaard Delta Rate (*BD-Rate*) [2], que mede o aumento de *bitrate* necessário para que uma técnica apresente a mesma qualidade visual que uma referência. Assim, valores de *BD-Rate* positivos são indesejáveis, pois indicam que uma técnica é menos eficiente do que a referência.

Os resultados de *speedup* e *BD-Rate* da técnica proposta, comparados aos resultados do particionamento uniforme, são apresentados na Tabela 1, que mostra os resultados agrupados por resolução do vídeo e o número de *Tiles* utilizados. A coluna 'Ganho' apresenta o aumento de *speedup* e a diferença de *BD-Rate* quando comparado ao particionamento uniforme. Nem todos os particionamentos puderam ser aplicados à todas as resoluções de vídeos, porque é necessário que os *Tiles* possuam um tamanho mínimo, que não pode ser atingido utilizando-se muitos *Tiles* em vídeos de resoluções pequenas. Analisando os resultados da Tabela 1, é possível perceber que a técnica desenvolvida foi capaz de aumentar o *speedup* dos particionamentos 2x2, 3x3, 4x4, e 5x5 em 3,2%, 10,7%, 8,2%, e 9,6%, respectivamente. Esses ganhos geram impactos mínimos em *BD-rate* de -0,04%, +0,08%, -0,07%, e +0,16% para os particionamentos 2x2, 3x3, 4x4, e 5x5, respectivamente.

Tabela 1. Resultados de *speedup* e *BD-Rate* da técnica desenvolvida vs. particionamento uniforme

| Tiles                      | Uniforme       |                | Desenvolvido   |                | Ganho          |                |
|----------------------------|----------------|----------------|----------------|----------------|----------------|----------------|
|                            | <i>Speedup</i> | <i>BD-Rate</i> | <i>Speedup</i> | <i>BD-Rate</i> | <i>Speedup</i> | <i>BD-Rate</i> |
| <b>Resolução 1920x1080</b> |                |                |                |                |                |                |
| 2x2                        | 3,49           | 0,68%          | 3,72           | 0,64%          | 7,1%           | -0,04          |
| 3x3                        | 7,00           | 1,43%          | 7,64           | 1,51%          | 9,6%           | 0,09           |
| 4x4                        | 11,51          | 2,27%          | 12,68          | 2,27%          | 11,0%          | 0,00           |
| 5x5                        | 16,60          | 3,19%          | 18,08          | 3,36%          | 9,6%           | 0,16           |
| <b>Resolução 1280x720</b>  |                |                |                |                |                |                |
| 2x2                        | 3,64           | 2,79%          | 3,69           | 2,70%          | 1,3%           | -0,09          |
| 3x3                        | 6,80           | 5,32%          | 7,71           | 5,40%          | 14,9%          | 0,08           |
| 4x4                        | 11,87          | 8,86%          | 12,45          | 8,73%          | 5,4%           | -0,13          |
| <b>Resolução 832x480</b>   |                |                |                |                |                |                |
| 2x2                        | 3,45           | 0,87%          | 3,48           | 0,88%          | 1,1%           | 0,02           |
| 3x3                        | 6,15           | 1,91%          | 6,61           | 2,00%          | 7,5%           | 0,09           |

A Figura 4 mostra os resultados de *speedup* ao longo do tempo para dois vídeos e particionamentos de *Tiles* distintos, onde o eixo horizontal representa os quadros do vídeo e o eixo vertical o valor de *speedup*. A Figura 4 (a) apresenta os resultados de *speedup* para os primeiros 40 quadros do vídeo *BasketballDrive*, com um particionamento de *Tiles* 2x2, enquanto que a Figura 4 (b) apresenta os resultados para os primeiros 40 quadros do vídeo *Johnny*, com um particionamento de *Tiles* 4x4. Observando estes gráficos, é possível perceber que existe um espaço significativo entre a linha que simboliza o *speedup* da técnica desenvolvida e o do

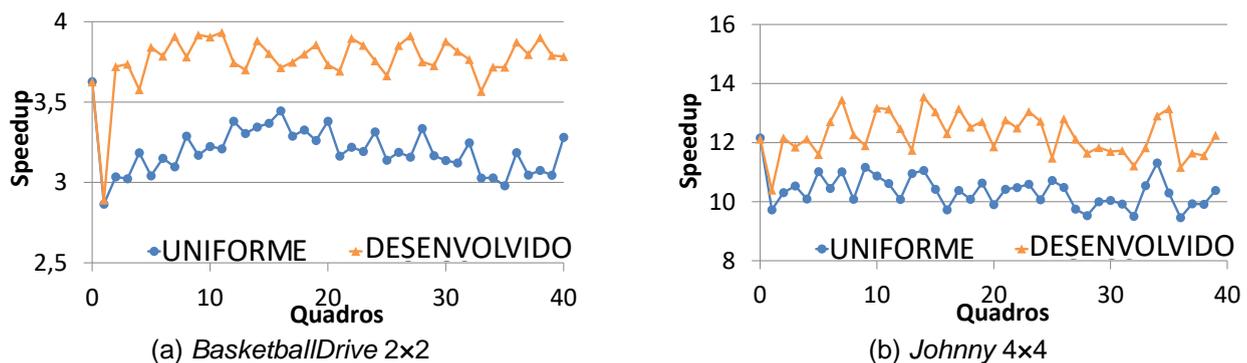


Figura 4. Resultados de *speedup* ao longo do tempo

particionamento uniforme. Além disso, as linhas raramente se encontram. Isso indica que a técnica desenvolvida além de propiciar um aumento de *speedup* médio, também mantém os ganhos para todos os quadros analisados.

Buscando comparar os resultados da técnica desenvolvida com os resultados de [1], a metodologia de avaliação dos autores foi replicada, isso é, foram utilizados os mesmos parâmetros de codificação e os mesmos vídeos foram codificados com o particionamento 2x2, o único apresentado em [1]. Por fim, os resultados foram avaliados utilizando o ATS para medir o paralelismo e *BD-Rate* para medir eficiência de codificação. Os resultados da técnica desenvolvida seguindo esta metodologia de avaliação e comparados com [1] são apresentados na Tabela 2. Nesta tabela, o *BD-Rate* do particionamento uniforme é usado como âncora e o *BD-Rate* das duas técnicas está representado de forma relativa ao do particionamento uniforme. Além do mais, como o particionamento uniforme é usado como referência, este possui um ATS de 0% em todos os casos, que foi omitido. A coluna 'Diferença' representa a diferença dos resultados da técnica desenvolvida quando comparada ao trabalho [1]. Analisando estes resultados, é possível perceber que a técnica desenvolvida apresenta um ATS superior ao do trabalho comparado, com um ganho médio de ATS igual a 1,53% e uma redução de *BD-Rate* média de 0,06%, ou seja, ganhando em ambos os eixos de comparação.

Tabela 2. Resultados de ATS e *BD-Rate* da técnica desenvolvida vs. Ahn [1]

| Resolução | Uniforme<br><i>BD-Rate</i> | Ahn [1] |                | Desenvolvido |                | Diferença |                |
|-----------|----------------------------|---------|----------------|--------------|----------------|-----------|----------------|
|           |                            | ATS     | <i>BD-Rate</i> | ATS          | <i>BD-Rate</i> | ATS       | <i>BD-Rate</i> |
| 1920x1080 | 0.64%                      | 6.10%   | -0.05%         | 6.31%        | -0.04%         | 0.21%     | 0.01%          |
| 832x480   | 0.98%                      | 1.45%   | +0.05%         | 4.29%        | -0.07%         | 2.84%     | -0.12%         |

### Conclusões:

Este trabalho apresentou uma nova técnica de particionamento de *Tiles* dinâmico, baseado em histórico de distribuição de carga de trabalho, para melhorar a eficiência do padrão de codificação de vídeo HEVC. Quando comparada ao particionamento uniforme padrão, a técnica desenvolvida atingiu um ganho de *speedup* médio de 7,9%, enquanto que a eficiência de codificação foi reduzida em 0,04%, ou seja, é possível atingir ganhos de *speedup* significativos com um impacto desprezível na eficiência de codificação. Quando comparado com outro trabalho da literatura, a técnica desenvolvida atingiu os melhores resultados, com ganhos de ATS em torno de 1,53% e melhorias na eficiência de codificação.

Sendo assim, é possível concluir que a técnica desenvolvida é capaz de proporcionar ganhos de paralelismo significativos no codificador HEVC enquanto mantém a eficiência de codificação inalterada. Assim, é possível otimizar o uso de recursos em dispositivos com múltiplas unidades de processamento que, além de reduzir o tempo de processamento, também leva a reduções no consumo energético destes dispositivos.

### Referências bibliográficas

- [1] AHN, Yong-Jo, et al. Complexity model based load-balancing algorithm for parallel tools of HEVC. Em: **Visual Communications and Image Processing (VCIP)**, 2013. IEEE, 2013. p. 1-5.
- [2] BJONTEGARRD, Gisle. Calculation of average PSNR differences between RD-curves. Document **VCEG-M33**, 2001.
- [3] BOSSEN, Frank. Common test conditions and software reference configurations. Document **JCTVC-L1100**. JCTVC, Genebra, Suíça, 2013.
- [4] BROSS, Benjamin. High efficiency video coding (HEVC) text specification draft 9 (SoDIS). Document **JCTVC-K1003**. JCT-VC, Xangai, China, 2012.
- [5] CORREA, Guilherme, et al. "Performance and computational complexity assessment of high-efficiency video encoders." **IEEE Transactions on Circuits and Systems for Video Technology** 22.12 (2012): 1899-1909.
- [6] FULDSETH, Arild, et al. Tiles. Document **JCTVC-E408**. JCT-VC, Genebra, Suíça, 2011.
- [7] McCANN, Ken, et al. High efficiency video coding (HEVC) test model 16 (HM 16) improved encoder description. Document **JCTVC-S1002**. JCT-VC Estrasburgo, França, 2014.
- [8] OHM, Jens-Rainer, et al. Comparison of the coding efficiency of video coding standards—including high efficiency video coding (HEVC). **IEEE Transactions on circuits and systems for video technology**, v. 22, n. 12, p. 1669-1684, 2012.
- [9] STORCH, Iago, et al. Avaliação do potencial máximo de speedup usando Tiles para compressão de vídeo paralela segundo o padrão HEVC. Em: **Simpósio Brasileiro de Telecomunicações e Processamento de Sinal**, 2016.