

USANDO ÁRVORES DE DECISÃO PARA ACELERAR A TRANSCODIFICAÇÃO DE VÍDEO H.266/VVC PARA AV1

CAROLINE SOUZA CAMARGO; ALEX BORGES; GUILHERME CORREA

Universidade Federal de Pelotas (UFPEL) – {caroline.sc, amborges, gcorrea}@inf.ufpel.edu.br

1. INTRODUÇÃO

No primeiro semestre de 2022, o consumo de vídeos na internet representou 65,93% do volume total de dados trafegados na internet, registrando um crescimento significativo de 24% em comparação com o mesmo período do ano anterior (SANDVINE, 2023). A operação de plataformas de *streaming* só é possível graças a algoritmos eficientes de compressão de vídeo, uma vez que vídeos não comprimidos exigem largura de banda inviável para a transmissão na internet. Os formatos de compressão de vídeo mais recentes no mercado, como *AOMedia Video 1* (AV1) (HAN et al., 2021) e *Versatile Video Coding* (VVC) (BROSS et al., 2019), atingem taxas de compressão superiores a 30% em comparação com *codecs* anteriores como o formato VP9 (FRAUNHOFER, 2020).

Além disso, a técnica de transcodificação de vídeo também é uma prática essencial para as plataformas de *streaming*, pois possibilita que vídeos sejam codificados para melhor se adequarem a diferentes dispositivos de reprodução e condições de rede para tráfego de dados (AHMAD et al., 2005). Outra motivação importante para a transcodificação está relacionada aos custos de *royalties* de patentes e modelos de licenciamento associados aos *codecs* de vídeo, como é o caso do VVC (ELTZROTH e CARY, 2021). Habilitar a transcodificação de vídeo para um formato de compressão livre de *royalties*, como o AV1, é recomendado e desejável para reduzir os custos das empresas de *streaming*.

No entanto, apesar dos codificadores VVC e AV1 oferecerem alta compressão de vídeo, eles consomem muitos recursos computacionais, aumentando o consumo energético. Ambos usam codificação baseada em blocos, com diferentes tamanhos e técnicas de otimização possíveis. No H.266/VVC, os quadros são divididos em unidades de árvore de codificação (CTUs), que podem ser subdivididas em unidades de codificação (CUs) menores usando uma árvore quaternária (QTMT) (HUANG et al., 2021). Já no AV1, os quadros são divididos em superblocos (SBs), com nove tipos diferentes de divisões possíveis (BROSS et al., 2019). Além disso, vários modos de predição intra, vetores de movimento e modos de transformação são avaliados para cada bloco candidato, o que aumenta a eficiência, mas também a complexidade do codificador.

Essa complexidade se torna ainda mais evidente na transcodificação de vídeo, uma vez que o vídeo previamente codificado em um formato precisa ser decodificado antes de passar pela etapa de codificação novamente. No entanto, percebe-se uma falta de soluções aceleradas de transcodificação de vídeo para *codecs* emergentes, pois, segundo os nossos conhecimentos, não existe uma solução de literatura publicada disponível de transcodificação de VVC para AV1. Portanto, este trabalho apresenta uma proposta para acelerar a transcodificação de vídeo de H.266/VVC para AV1. A ideia consiste em usar dados obtidos durante a fase de particionamento dos blocos, tanto do processo de decodificação como de recodificação. Os dados obtidos serão usados para treinar modelos baseados em aprendizado de máquina que determinam se o processo de sub-particionamento deve ser habilitado ou não durante a codificação.

2. METODOLOGIA

Um processo de decodificação foi realizado em vídeos inicialmente codificados em H.266/VVC para obter dados brutos e permitir a criação de modelos preditivos. Esses vídeos foram posteriormente codificados no formato AV1, usando o vídeo decodificado como fonte. Assim, as informações extraídas foram organizadas em duas matrizes, na qual cada célula representa a menor partição de bloco permitida em H.266/VVC e AV1. Durante a codificação de cada novo superbloco, essas matrizes são atualizadas com novos dados e usadas em conjunto como entrada para um modelo de aprendizado de máquina. Esse modelo é treinado para determinar se o bloco atual deve ser particionado ou não.

Os dados de entrada são obtidos de cinco regiões vizinhas de um bloco (R1, R2, R3, R4 e R5) tanto do quadro do decodificador quanto do quadro do codificador, conforme mostrado na Figura 1. Na figura, o bloco em processamento de tamanho $N \times N$ pixels requer uma decisão do modelo, ou seja, se deve ou não ser subdividido. Um total de 25 características são consideradas, cinco para cada vizinho: (i) a profundidade da árvore de particionamento, (ii) o tamanho do bloco, (iii) a direção de particionamento (ou seja, vertical ou horizontal), (iv) o modo de predição e (v) o tipo de predição (ou seja, intra quadros ou inter quadros).

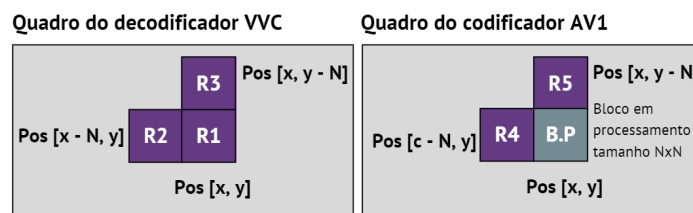


Figura 1. Regiões vizinhas para capturar dados para uso nos modelos de aprendizado de máquina (coletados para cada bloco em processamento)

Para permitir a decisão se o bloco atual deve ser ou não subdividido foi necessário modificar o fluxo de execução de particionamento de blocos no *software* de referência *libaom* para incluir os processos de antecipação do modo de divisão *SPLIT*. Assim, um total de 12 modelos preditivos foram treinados, cada um para uma combinação específica de valores de Qualidade Constante (CQ), conforme recomendado pela AOMedia (20, 32, 43 e 55) (DAEDE et al., 2020), e os três primeiros blocos quadrados maiores no AV1 (128×128, 64×64 e 32×32).

Neste trabalho, o *software* de referência para o padrão H.266/VVC (VTM, versão 19.0, hash c71f7a9e) e o formato AV1 (*libaom*, versão 3.5, código hash 9a83c6) foram usados para realizar os experimentos e a coleta de dados. Para possibilitar a transcodificação, o vídeo original foi codificado no formato VVC com um parâmetro de quantização (QP) igual a 22, em seguida, decodificado para gerar o vídeo intermediário a ser transcodificado para o AV1 com e sem aceleração. Todos os experimentos foram realizados nos primeiros 60 quadros de três grupos de sequências de vídeos recomendadas para testes. Sete sequências de resolução 1920×1080 foram usadas para treinamento, outras sete na mesma resolução foram usadas para teste, e nove sequências em 1280×720 e 13 sequências em 1920×1080 foram usadas para a etapa de predição.

O algoritmo de Árvores de Classificação e Regressão (CART), disponível no pacote *SciKit-Learn* do Python, foi escolhido para treinar todos os modelos preditivos apresentados neste trabalho. Após a conclusão do processo de treinamento foram geradas todas as combinações de hiperparâmetros e no total foram gerados 9.526.572 modelos candidatos. Esse número é inviável para

realização correta de treinamentos e testes, portanto, o algoritmo *HalvingRandomSearchCV* foi usado para realizar uma análise de todos os candidatos em um modo exaustivo.

Após a aplicação deste algoritmo, restaram 229 candidatos que passaram por treinamento e teste com sequências apropriadas. Em seguida, um filtro foi utilizado para descartar modelos com valor de área sob a curva ROC (AUC) abaixo de 0,5. A melhor combinação de hiperparâmetros encontrada incluiu *Criterion* 'entropy', *Splitter* 'best', *Max Depth* 11, *Min Samples Split* 11, *Min Samples Leaf* 11, *Max Features* 'sqrt', *Max Leaf Nodes* 7, *Min Impurity Decrease* 0,6 e *CPP Alpha* 0,5. Com esses hiperparâmetros, os 12 modelos obtiveram uma pontuação média de F1 de 96,25%. Todos os modelos treinados para o tamanho de bloco 128x128 alcançaram F1 acima de 99%, enquanto para o tamanho de bloco 32x32, a pontuação de F1 atingiu 95%.

3. RESULTADOS E DISCUSSÃO

A Tabela 1 apresenta os resultados, no qual o *BD-rate* foi utilizado para avaliar a eficiência de codificação e o tempo de codificação foi empregado para calcular a aceleração de transcodificação em relação ao codificador libaom não alterado. As sequências de vídeo listadas são aquelas reservadas para a predição. A tabela mostra que a aceleração proposta alcança uma redução média de 12,60% no tempo de codificação do AV1. Essa aceleração vem com o custo de um aumento médio na *BD-rate* de 1,81%.

Vídeos	BD-Rate (%)	Time Saving (%)	Vídeos	BD-Rate (%)	Time Saving (%)
<i>boat_hdr amazon</i>	0,3	5,42	<i>Netflix Crosswalk</i>	2,08	6,65
<i>dark</i>	6,13	28,2	<i>Netflix FoodMarket</i>	2,75	8,51
<i>FourPeople</i>	1,11	7,08	<i>Netflix PierSeaside</i>	1,62	12,73
<i>Johnny</i>	1,59	33,49	<i>Netflix TunnelFlag</i>	2,18	20,67
<i>KristenAndSara</i>	1,01	9,65	<i>pan_hdr amazon</i>	0,23	0,05
<i>Netflix DinnerScene</i>	2,74	2,38	<i>park joy</i>	0,4	13,85
<i>Netflix RollerCoaster</i>	1,84	14,84	<i>pedestrian area</i>	2,44	7,35
<i>Netflix Tango</i>	1,77	10,73	<i>rush field cuts</i>	0,64	16,46
<i>Vidyo4</i>	1,3	11,59	<i>rush hour</i>	3,41	25,63
<i>aspen</i>	4,03	18,02	<i>seaplane_hdr amazon</i>	0,13	2,93
<i>ducks take off</i>	1,91	20,24			
<i>guitar_hdr amazon</i>	0,22	0,74	MÉDIA	1,81	12,6

Tabela 1. Eficiência de codificação e resultados da economia de tempo

Essa economia de tempo se torna particularmente vantajosa ao lidar com um grande volume de vídeos sendo transcodificados em diversos formatos, qualidades e resoluções. Por outro lado, como esperado, o aumento observado na *BD-rate* pode levar a uma degradação na qualidade final do vídeo, apesar de ser um valor relativamente baixo. Analisando os resultados em particular, pode-se destacar as sequências “*dark*”, “*aspen*” e “*rush hour*”, que exibem a maior taxa de *BD-rate*. Certos atributos presentes nesses vídeos incluem texturas mais complexas, como folhas de árvores, movimentos de carros e variações na iluminação e contraste. Assim, a hipótese para esses resultados é que o modelo de ML implementado não se adaptou bem às características das sequências

mencionadas, pois fatores como variações de textura, contraste, iluminação e sombras, por exemplo, são características que não foram levadas em consideração pelo modelo durante o processo de re-codificação.

Além disso, vídeos como "*Johnny*" e "*ducks take off*", mostraram melhorias notáveis, com uma redução de *BD-rate* inferior a 2% e uma aceleração de mais de 20%, sugerindo que o modelo de aprendizado de máquina otimizou a codificação sem comprometer muito a qualidade visual. Vale ressaltar, que não há outras propostas conhecidas na literatura para acelerar a transcodificação de H.266/VVC para AV1, tornando comparações diretas difíceis.

4. CONCLUSÕES

H.266/VVC e AV1 são o estado da arte atual em padrões de codificação de vídeo e formato na indústria de multimídia e *streaming*. Por questões de compatibilidade e também devido às políticas de *royalties*, migrar de H.266/VVC para AV1 é uma tarefa importante para empresas de *streaming*. Este trabalho apresentou uma solução baseada em aprendizado de máquina que acelera a transcodificação de H.266/VVC para AV1 por meio do uso de 12 modelos de árvores de decisão treinados com o algoritmo *Classification and Regression Trees* (CART). Com base nas informações coletadas durante o processo de decodificação de H.266/VVC, os modelos inferem se um bloco deve ser dividido durante o processo de re-codificação AV1, evitando avaliações desnecessárias. Quando implementada no *software* de referência libaom, a solução alcançou uma diminuição de 12,60% no tempo de re-codificação do AV1, com perdas de eficiência de codificação de 1,81% em média. Até onde os autores sabem, esta é a primeira solução focada na aceleração do transcodificador H.266/VVC para AV1.

5. REFERÊNCIAS BIBLIOGRÁFICAS

- AHMAD, I. et al. Video transcoding: **an overview of various techniques and research issues**. IEEE Transactions on Multimedia, 2005. doi: 10.1109/TMM.2005.854472.
- BROSS, B. et al. **Versatile video coding** (draft 7), outubro de 2019.
- DAEDE, T. et al. **Video codec testing and quality measurement**, janeiro de 2020. Online. Disponível em: <https://tools.ietf.org/html/draft-ietf-netvc-testing-09>.
- ELTZROTH, C. e CARY, J. **Fostering of patent pools covering cable technology: Lessons from vvc pool fostering**, outubro de 2021. doi: 10.2139/ssrn.3949545.
- FRAUNHOFER. **Fraunhofer HHI is proud to present the new state-of-the-art in global video coding: H.266/VVC brings video transmission to new speeds**, julho de 2020. Online. Disponível em: <https://newsletter.fraunhofer.de/viewonline2/17386/465/19/6RFhct0v/7BvSjdE9tM/1>.
- HAN, J. et al. **A technical overview of av1**. Proceedings of the IEEE, 2021. doi: 10.1109/JPROC.2021.3058584.
- HUANG, Y.-W. et al. **Block partitioning structure in the vvc standard**. IEEE Transactions on Circuits and Systems for Video Technology, 2021. doi: 10.1109/TCSVT.2021.3088134.
- SANDVINE. **Global internet phenomena**, janeiro de 2023. Online. Disponível em: <https://www.sandvine.com/phenomena>.