

DECISÃO DE PARTICIONAMENTO DE BLOCOS NO PADRÃO DE CODIFICAÇÃO DE VÍDEO VVC COM APRENDIZADO DE MÁQUINA

MATHEUS LINDINO¹; MATEUS GRELLERT²; BRUNO ZATT¹;
GUILHERME CORREA¹

¹Universidade Federal de Pelotas – Video Technology Research Group

²Universidade Federal de Santa Catarina – Embedded Computing Laboratory

¹{mclindino, zatt, gcorrea}@inf.ufpel.edu.br, ²mateus.grellert@ufsc.br

1. INTRODUÇÃO

No cenário mundial atual, com a quarentena em diversos países devido à pandemia de COVID-19, a demanda por serviços de *streaming* de vídeos cresceu significativamente. Com o surgimento de novas tecnologias, por exemplo, *Ultra High-Definition* (UHD) e *Virtual Reality* (VR), a quantidade de dados necessários para representar mídias digitais cresce devido ao aumento de informação presente.

Neste sentido, o *High Efficiency Video Coding* (HEVC) (SULLIVAN, 2012) já não consegue comprimir com eficácia esses tipos de mídia. Para isso, desde 2015, um novo padrão de codificação de vídeo denominado *Versatile Video Coding* (VVC) (BROSS, 2018) passou a ser desenvolvido pelo *Moving Picture Experts Group* (MPEG). Este padrão, possuindo novos algoritmos e maior flexibilidade para tratamento de diferentes tipos de conteúdo, se mostrou mais competente em questão de eficiência de codificação que seu antecessor. Todavia, ele acaba se tornando computacionalmente mais complexo. Assim, estratégias de otimização, que possuam objetivo de diminuir a complexidade sem alterar a eficiência de codificação são extremamente necessárias para o VVC.

Neste contexto, este trabalho apresenta a exploração de soluções para redução de custo computacional do codificador VVC com uso de aprendizado de máquina. A solução é aplicada especificamente à decisão de particionamento do VVC, que consiste em uma ferramenta flexível que permite adaptação e alto grau de liberdade para o codificador realizar a codificação com unidades de dados de diferentes formatos e tamanhos, adaptando-se ao conteúdo em questão.

2. REFERENCIAL TEÓRICO

Uma das ferramentas novas que o VVC introduziu foram as *QuadTree* (QT) e *Multi-type Tree* (MT). Estas estruturas são usadas na decisão de particionamento do quadro do vídeo em blocos menores. Esse particionamento ocorre da seguinte maneira: primeiro, todo o quadro é dividido em blocos de tamanhos iguais de 128x128 pixels, chamados de *Coding Tree Unit* (CTU). Em seguida, as CTUs são particionadas pela QT ou pela MT, gerando blocos de tamanhos variados chamados *Coding Units* (CUs). Utilizar um particionamento QT leva a quatro novas CUs de tamanhos quadrados. No caso do particionamento MT, quatro tipos de particionamentos podem ser realizados: binário vertical (BV), binário horizontal (BH), ternário vertical (TV) e ternário horizontal (TH), conforme apresentado na Fig. 1. Para simplificar o processo de particionamento, após o processo de MT começar, não é permitido realizar um particionamento do tipo QT. Também, o codificador deve determinar o modo de predição que será utilizado para aquela CU, podendo ser intra-quadro (que explora redundância espacial) ou inter-quadro (que explora redundância temporal). Em um codificador ótimo, a

decisão do melhor particionamento é complexa, uma vez que o codificador testará todas as combinações de QT e MT existentes de particionamentos.

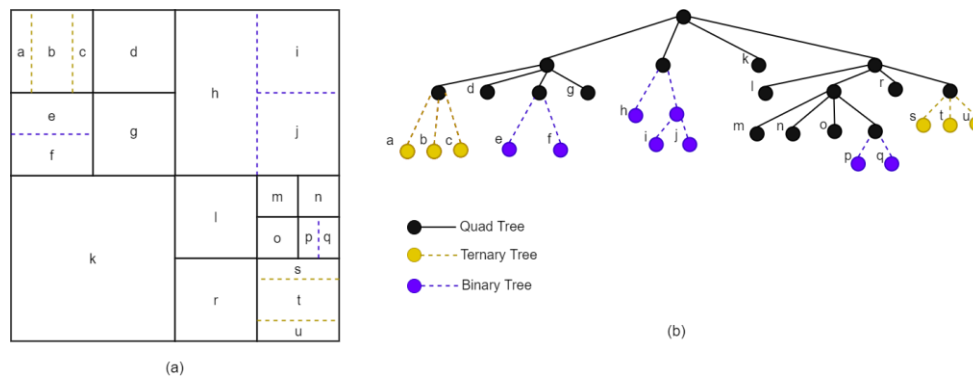


Fig 1. Ilustração da estrutura QT e MT: (a) particionamento de uma CTU e (b) estrutura da árvore correspondente.

Para simplificar os testes complexos, uma abordagem possível é utilizar aprendizado de máquina para identificar as necessidades de particionamento de blocos. Neste caso, os *Random Forests* (RF) são uma classe de algoritmos frequentemente utilizados. Em comparação com as árvores de decisão, RFs são menos propensas a sofrer *overfitting*, pois são compostas por várias árvores de decisão construídas sistematicamente pela seleção pseudoaleatória de subconjuntos de recursos (KAM HO, 1998). Desta forma, se um conjunto relevante de variáveis de entrada é usado no processo de treinamento da RFs, estas podem alcançar alta precisão ao custo de uma pequena sobrecarga em termos de uso de recursos computacionais.

Na literatura, existem escassez de trabalhos que buscam acelerar os processos do codificador VVC. (YANG, 2019) propõe um algoritmo para diminuir a complexidade da estrutura QT e MT, no modo intra-quadro do codificador. (WIECKOWSKI, 2019) traz diferentes abordagens para diminuir o tempo de codificação do VVC. (PARK, 2019) demonstra um novo método para otimizar o processo de *Affine Motion Estimation* (AME). Por conseguinte, este trabalho propõe aperfeiçoar a decisão de particionamento do bloco em predição inter-quadro e em intra-quadro, utilizando aprendizado de máquina.

3. METODOLOGIA

Para a execução deste trabalho, foram extraídas 51 características de cada bloco no momento de decisão do particionamento, a fim de estabelecer um conjunto de dados e uma coluna de *targets* (valores de saída esperados da RFs). Este conjunto será utilizado para a criação e treinamento dos modelos preditivos. Para isso, foram retirados vídeos da base de dados do *Ultra Video Group* (UVG) e calculado os valores de *Spatial Index* (SI) e *Temporal Index* (TI) (ITU-R, 2008), totalizando 15 vídeos selecionados. Eles foram codificados em quatro QPs (22, 27, 32, 37) com o software de referência VTM 9.0 (VTM).

Para garantir um melhor desempenho e precisão do algoritmo, os dados foram balanceados. Cada vídeo, QP, tamanho de bloco e *targets* foi padronizado com a mesma quantidade de exemplos, utilizando as técnicas de *Random Undersampling* e *Random Oversampling*.

Na sequência, esse conjunto foi dividido em três partes: 1) Conjunto de treinamento, 2) Conjunto de validação, e 3) Conjunto de teste. Também foram determinados nove modelos, um para cada tamanho de bloco e configuração de

codificador de vídeo. A seguir, foi verificada a importância de cada característica do bloco em relação à decisão de particionamento (Cf. Fig. 2).

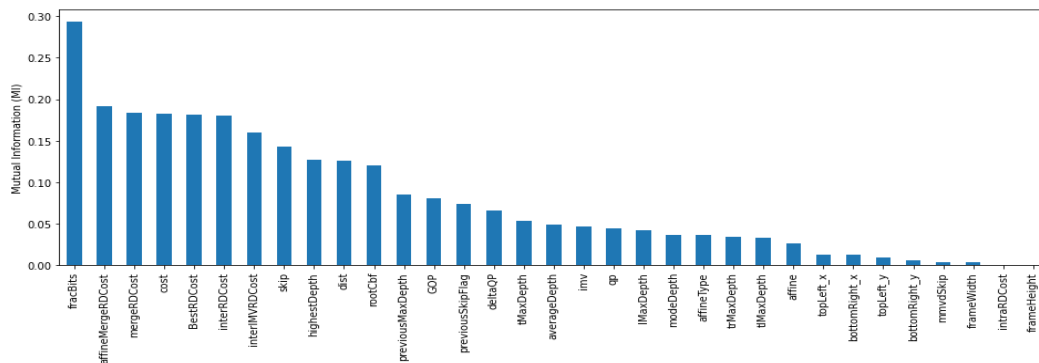


Fig. 2. Ganho de informação de *features* do modelo 128x128, *Random Access*.

Optou-se pelas características com maior ganho de informação para cada modelo, desta forma, simplificando as RFs.

Findada esta etapa, iniciou-se o processo de treinamento. Como a RF tem vários parâmetros e ajustar cada um acaba se tornando uma tarefa Árdua, optou-se por calcular o coeficiente de correlação de Pearson (Cf. Tabela 1) para cada hiperparâmetro.

Tabela 1. Coeficiente de correlação de Pearson na configuração *Random Access*.

Hiperparâmetros							
modelo	n_estimators	min_samples_split	min_samples_leaf	max_features	max_depth	bootstrap	max_samples
128x128	0,077	-0,0268	-0,1365	0,076	0,5527	-0,0814	0,0621
64x64	0,0911	-0,0536	-0,0055	0,2562	0,6208	-0,0432	-0,0091
32x32	0,1422	0,0134	0,0184	0,2223	0,4203	0,0124	0,0213
16x16	0,1407	0,0713	0,0388	0,1116	0,4676	0,0669	0,0051
8x8	0,2643	-0,0322	0,0102	0,0695	0,3453	0,1258	0,1657

Desde coeficiente, foi selecionado os dois hiperparâmetros que mais impactam na precisão de cada modelo. A partir deles, foi realizado uma busca exaustiva com o intuito de garantir o melhor valor de cada hiperparâmetro. Esta busca promoveu também a melhor precisão.

4. RESULTADOS E DISCUSSÃO

Para a validação dos modelos, foi utilizado o conjunto de validação e teste assegurando que o algoritmo não sofra de *overfitting*. Foi realizada a técnica de validação cruzada, com cinco *folds*, a fim de avaliar a capacidade de generalização de um determinado modelo. Os resultados estão em F1-score, uma medida que mensura a precisão de modelos para classificação binária.

A partir da técnica, foram obtidos os seguintes resultados, apresentados na Tabela 2.

Tabela 2. Resultados de cada modelo treinado.

F1-score (%)					
Modelo	128x128	64x64	32x32	16x16	8x8
Random Access	85,83	87,98	78,05	75,14	76,76
All Intra	-	93,59	79,04	74,77	80,03

Nota-se que a maioria dos modelos obtiveram uma precisão superior a 75%, visto que o algoritmo consegue decidir corretamente se um determinado bloco necessita ou não ser particionado, diminuindo o tempo e custo. Pode-se afirmar que esses valores ratificam a eficácia na decisão de particionamento para a redução de complexidade do codificador VVC e a alta precisão indica que terá baixo impacto de sobrecarga no codificador.

5. CONCLUSÕES

O presente trabalho apresentou um conjunto de modelos preditivos sob a forma de RFs, visando acelerar a decisão de particionamento do codificador VVC sem impactar na eficiência de codificação. Constatou-se que os modelos conseguem identificar blocos que necessitam ou não continuar o seu particionamento, uma tarefa árdua para o codificador padrão, com o máximo de 93,59% de precisão.

Por fim, conclui-se que o método apresentando sugere a redução de tempo de codificação e a diminuição de recursos computacionais e sugere também estudos sobre implementação dos modelos no software de referência, para mensurar a redução de tempo em relação ao codificador padrão VVC.

6. REFERÊNCIAS BIBLIOGRÁFICAS

- G. J. Sullivan, J. Ohm, W. Han and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1649-1668, Dec. 2012, doi: 10.1109/TCSVT.2012.2221191.
- B. Bross, "Versatile Video Coding (Draft 1)", JVET-J1001, Joint Video Exploration Team (JVET), Apr. 2018.
- H. Yang, L. Shen, X. Dong, Q. Ding, P. An and G. Jiang, "Low-Complexity CTU Partition Structure Decision and Fast Intra Mode Decision for Versatile Video Coding," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 6, pp. 1668-1682, June 2020, doi: 10.1109/TCSVT.2019.2904198.
- A. Wieckowski, J. Ma, H. Schwarz, D. Marpe and T. Wiegand, "Fast Partitioning Decision Strategies for The Upcoming Versatile Video Coding (VVC) Standard," 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 2019, pp. 4130-4134, doi: 10.1109/ICIP.2019.8803533.
- S. Park, and J. Kang, "Fast Affine Motion Estimation for Versatile Video Coding (VVC) Encoding," in IEEE Access, vol. 7, pp. 158075-158084, 2019, doi: 10.1109/ACCESS.2019.2950388.
- Tin Kam Ho, "The random subspace method for constructing decision forests," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 832-844, Aug. 1998, doi: 10.1109/34.709601.
- Versatile Video Coding Test Model(VTM) 9.0 :
https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/tree/VTM-9.0
- ITU-R P.910, "Subjective Video Quality Assessment Methods for Multi-Media Applications," (2008).