

DECISÃO RÁPIDA DE FILTROS DE INTERPOLAÇÃO USANDO APRENDIZADO DE MÁQUINA PARA A PREDIÇÃO INTER FRACIONÁRIA NO AV1

PATRICK SILVA DA ROSA¹; DAIANE FONSECA FREITAS²;
LEONARDO LUÍS MULLER³; GUILHERME RIBEIRO CORREA⁴;
DANIEL MUNARI VILCHEZ PALOMINO⁵

¹Universidade Federal de Pelotas – psdrosa@inf.ufpel.edu.br

²Universidade Federal de Pelotas – dffreitas@inf.ufpel.edu.br

³Universidade Federal de Pelotas – llmuller@inf.ufpel.edu.br

⁴Universidade Federal de Pelotas – gcorrea@inf.ufpel.edu.br

⁵Universidade Federal de Pelotas – dpalomino@inf.ufpel.edu.br

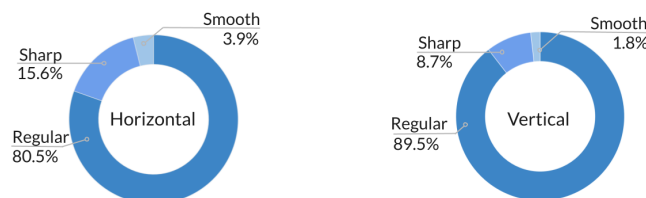
1. INTRODUÇÃO

O consumo de conteúdo audiovisual pela internet tem crescido de forma acelerada, impulsionado por *streaming*, redes sociais e transmissões ao vivo. No final de 2024, o tráfego de vídeo representava 74% de todo o tráfego de dados móveis, reflexo da popularização de dispositivos de alta definição e conexões rápidas (Ericsson, 2025). Esse cenário exige soluções capazes de processar e transmitir grandes volumes de dados sem perda perceptível de qualidade.

O codec AV1, desenvolvido pela Alliance for Open Media, oferece até 40% de economia de bitrate em relação ao HEVC, mantendo a qualidade visual (AOMedia, 2018). Essa eficiência é viabilizada por ferramentas avançadas de codificação, como compensação de movimento aprimorada e filtros de interpolação adaptativos. No entanto, esses recursos elevam a complexidade computacional, especialmente na etapa de *Fractional Motion Estimation* (FME).

A FME refina vetores de movimento para posições subpixel, usando filtros das famílias Regular, Smooth e Sharp, que podem ser combinados de nove formas possíveis (Han et al., 2020). Essa flexibilidade aumenta a eficiência, mas também o custo, pois cada combinação exige cálculo de custo taxa-distorção (RD-Cost). Em testes feitos mostra que o filtro Regular é usado em mais de 80% dos casos na horizontal e quase 90% na vertical como mostra a **Figura 1**, motivando a criação de uma abordagem que prioriza essas combinações para reduzir o tempo de processamento.

Figura 1. Escolha dos filtros de interpolação em ambas direções



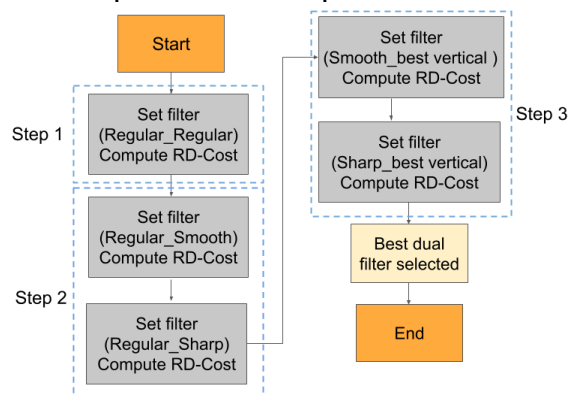
2. METODOLOGIA

Para acelerar o processo definido no software de referência do AV1 (libaom), mostrado na **Figura 2**, utiliza-se uma busca em três etapas.

No primeiro passo, aplica-se a combinação Regular_Regular e calcula-se o

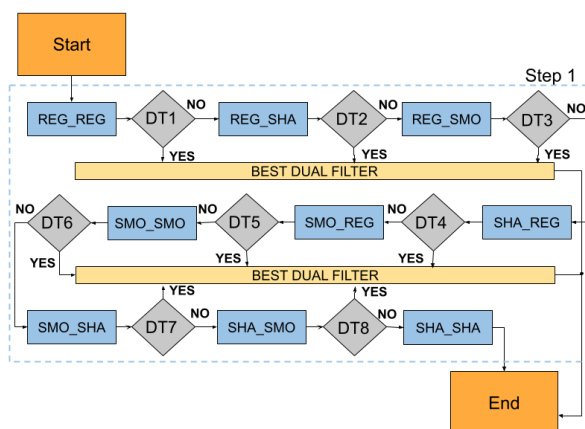
RD-Cost. No segundo passo, o filtro na direção vertical é variado, avaliando as combinações Regular_Smooth e Regular_Sharp, com o cálculo do RD-Cost para cada uma, a fim de selecionar o melhor filtro vertical. Por fim, no terceiro passo, calculam-se os RD-Costs das combinações Smooth_BestVertical e Sharp_BestVertical. Ao término desse processo, escolhe o filtro com o menor custo de taxa-distorção.

Figura 2. Processo rápido de busca para filtros de interpolação no libaom.



A solução proposta utiliza oito árvores de decisão, cada uma responsável por avaliar uma combinação de filtros, priorizando as mais frequentes. O processo inicia com REG_REG, como visto na **Figura 3**; se validada, encerra-se a busca. Caso contrário, as próximas árvores são consultadas até selecionar a melhor combinação, evitando as três etapas sequenciais do *fast search* padrão.

Figura 3. Processo modificado de busca de filtros de interpolação



O conjunto de dados foi extraído do encoder AV1 (libaom), abrangendo vídeos HD, FHD e UHD, com 28 *features* diretamente relacionadas à codificação, como tamanho do bloco, posição e custo de predição. Para balancear as classes, aplicou-se o algoritmo SMOTE, gerando exemplos sintéticos das combinações menos frequentes.

O treinamento foi feito com o *DecisionTreeClassifier* do Scikit-learn, usando 80% dos dados para treino/validação e 20% para teste. A seleção de hiperparâmetros combinou *Random Search* e *Grid Search*, com avaliação pelo F1-score ponderado. As árvores apresentaram profundidade entre 13 e 16 e F1-scores de 78,77% a 90,67%, como mostra a **Tabela 1**, com melhor desempenho para combinações dominantes como REG_REG.

Tabela 1. F1-scores no modelo final após GridSearch

Árvore de Decisão	# de nós	Profundidade	F1-score
DT1_REG_REG	756	16	90.67%
DT2_REG_SHA	645	14	88.43%
DT3_REG_SMO	669	15	84.78%
DT4_SHA_REG	578	13	79.43%
DT5_SMO_REG	729	14	80.12%
DT6_SMO_SMO	491	15	82.22%
DT7_SMO_SHA	535	13	78.77%
DT8_SHA_SMO	572	14	79.04%

3. RESULTADOS E DISCUSSÃO

Os testes foram realizados com a libaom 3.10.0 em um sistema Intel i7-3770, 16 GB de RAM e Ubuntu 20.04.5 LTS, totalizando 200 codificações de 25 vídeos em três resoluções. As métricas usadas foram *Time Saving* (TS) e BD-BR (Bjøntegaard, 2001).

Tabela 2. Redução do tempo de codificação e eficiência de codificação

Resolução	BD-BR	Time Saving
High Definition	0.08%	2.11%
Full High Definition	0.14%	2.44%
Ultra High Definition	0.11%	2.70%
Média	0,11	2,42

Como pode ser visto na **Tabela 2**, a proposta obteve redução média de 2,42% no tempo de codificação e aumento médio de apenas 0,11% no BD-BR, mantendo a qualidade.

Tabela 3. Comparação com trabalhos relacionados

Solução	Software	Time Saving	BD-BR
Domanski	2.0.0	-	0.54%
Freitas	3.10.0	-	0.64 and 1.11%
Kolodziejski	3.9.2	3.38%	0.14%
Nossa	3.10.0	2.42%	0.11%

Comparando com trabalhos anteriores, o método mostrou bom equilíbrio entre aceleração e preservação da qualidade. Tanto Domanski (Domanski,2021) quanto Freitas (Freitas, 2024) são soluções em Hardware para acelerar a FME dificultando assim a comparação direta, entretanto tem BD-BR maior que a solução proposta. Já a única solução em software de Kolodziejski (Kolodziejski,

2025) obteve 3,38% de TS e 0,14% de BD-BR, porém em versão anterior do encoder e utilizando XGBoost. Frente a propostas de otimização em hardware, a nossa abordagem se destaca por ser totalmente em software e fácil de integrar ao fluxo atual do AV1.

4. CONCLUSÕES

Este trabalho apresentou uma Decisão de Filtro de Interpolação Rápida baseada em Aprendizado de Máquina para Estimativa de Movimento Fracionário AV1, utilizando árvores de decisão. Os resultados experimentais mostraram uma pequena, porém consistente redução no tempo de codificação com mínima perda de eficiência.

Além disso, o método proposto demonstrou uma melhoria robusta no processo de seleção, otimizando a busca pela melhor combinação de filtros de interpolação. Pesquisas futuras podem explorar técnicas complementares, como estratégias de terminação antecipada para decidir se deve ou não aplicar FME, a fim de acelerar ainda mais a busca pelo filtro de interpolação e reduzir a complexidade do codec AV1.

5. REFERÊNCIAS BIBLIOGRÁFICAS

BJONTEGAARD, G. **Calculation of average PSNR differences between RD-curves**. VCEG Meeting, https://www.itu.int/wftp3/av-arch/video-site/0104_Aus/VCEG-M33.doc, 2001.

Ericsson. (s.d.). *Ericsson Mobility Report – past and present reports*. Acessado em 18 de agosto de 2025, de <https://www.ericsson.com/en/reports-and-papers/mobility-report/reports>

J. Han, B. Li, D. Mukherjee, C.-H. Chiang, C. Chen, H. Su, S. Parker, U. Joshi, Y. Chen, Y. Wang, P. Wilkins, Y. Xu, and J. Bankoski, “**A technical overview of AV1**,” 2020.

A. for Open Media. (2018) **Alliance for Open Media**. [Online]. Available: <https://aomedia.googlesource.com/aom>

R. Domanski, W. Kolodziejcki, G. Correa, M. Porto, B. Zatt, and L. Agostini, “**Low-Power and High-Throughput Approximated Architecture for AV1 FME Interpolation**,” in Proc. IEEE International Symposium on Circuits and Systems (ISCAS), 2021, pp. 1–5.

D. Freitas, P. Rosa, L. Müller, D. Palomino, C. M. Diniz, M. Grellert, and G. Correa, “**Low-Power Multiversion Interpolation Filter Accelerator with Hardware Reuse for AV1 Codec**,” IEEE Journal on Emerging and Selected Topics in Circuits and Systems, 2024.

W. Kolodziejcki, M. Porto and L. Agostini, “**FIFS: A Machine Learning-Based Fast AV1 Interpolation Filter Search**,” 2025 IEEE 16th Latin America Symposium on Circuits and Systems (LASCAS), Bento Gonçalves, Brazil, 2025, pp. 1-5, doi: 10.1109/LASCAS64004.2025.10966280