

# ACELERAÇÃO DO TEST ZONE SEARCH E DA ESTIMAÇÃO DE MOVIMENTO AFFINE PARA O CODIFICADOR VVC USANDO APRENDIZADO DE MÁQUINA

RAMIRO VIANA<sup>1</sup>; MARCELO PORTO<sup>2</sup>;  
GUILHERME CORRÊA<sup>3</sup>; LUCIANO AGOSTINI<sup>4</sup>

<sup>1</sup>Universidade Federal de Pelotas (UFPel) – rgsviana@inf.ufpel.edu.br

<sup>2</sup>Universidade Federal de Pelotas (UFPel) – porto@inf.ufpel.edu.br

<sup>3</sup>Universidade Federal de Pelotas (UFPel) – gcorrea@inf.ufpel.edu.br

<sup>4</sup>Universidade Federal de Pelotas (UFPel) – agostini@inf.ufpel.edu.br

## 1. INTRODUÇÃO

A ampla utilização de conteúdo multimídia na sociedade atual transformou as dinâmicas de trabalho, educação e entretenimento. O trabalho remoto, o ensino à distância e *streaming* de filmes e séries de televisão são uma tendência irreversível na vida de todos. Isso é evidenciado pelo fato de que o número global de usuários da internet superou 5 bilhões em 2023, com aproximadamente 90% do tráfego da internet sendo atribuído a vídeos digitais (CISCO, 2023). Dada a complexidade de transmitir e armazenar vídeos digitais, o desenvolvimento de codificadores de vídeo é crucial para comprimir dados enquanto preserva a qualidade da imagem, embora isso exija recursos computacionais substanciais.

O *Versatile Video Coding* (VVC) é a tecnologia de codificação de vídeo mais avançada atualmente. Apesar de suas exigências computacionais substanciais, o VVC se destaca por suas taxas de compressão excepcionais, superando outros codificadores comerciais em eficiência, graças às suas ferramentas e algoritmos avançados que permitem uma compressão de vídeo altamente eficiente (LOOSE et al., 2022). A predição Inter-Quadros do VVC, responsável pela utilização da redundância temporal para melhor prever os quadros sendo codificados, incorpora ferramentas-chave para otimizar a codificação de vídeo, explorando a redundância espacial, como o *Test Zone Search* (TZS) e a Estimação de Movimento *Affine* (AME). Tanto o TZS quanto a AME utilizam 12 tamanhos de Blocos de Codificação, do inglês *Coding Blocks* (CBs), entre 16x16 e 128x128.

O *Test Zone Search* é conhecido por sua velocidade e eficiência no processo de Estimação de Movimento (ME) e é o algoritmo de correspondência de blocos mais utilizado nos codificadores de vídeo atuais. No entanto, apesar de sua rápida execução, o TZS continua sendo um dos algoritmos mais demorados no padrão VVC. O TZS consiste em quatro etapas: *Motion Vector Prediction*, *First Search*, *Raster Search* e *Refinement*. Sendo que a etapa do *Motion Vector Prediction* sempre precisa ser executada pelo codificador (VIANA et al., 2024).

A Estimação de Movimento *Affine* é crucial para aprimorar a Predição Inter-Quadros, estimando movimentos complexos, como redimensionamento, rotação e cisalhamento. No entanto, seu uso aumenta significativamente as demandas computacionais, resultando em um tempo total de codificação maior no processo de predição. O VVC emprega dois estágios de AME para mapear esses movimentos: o modelo de 4-Parâmetros para movimentos simples, como redimensionamento e rotação, e o modelo de 6-Parâmetros para movimentos mais complexos, como ajustes de tela e cisalhamento (LOOSE et al., 2022).

O aumento do esforço computacional exigido pelos codificadores de vídeo atuais impulsionou o desenvolvimento de novas soluções para reduzir esse esforço com baixo impacto na eficiência da codificação. Desta forma, o Aprendizado de

Máquina, do inglês *Machine Learning* (ML), se tornou uma ferramenta crucial. Entre os modelos de ML, as Árvores de Decisão, do inglês *Decision Trees* (DT) se destacam por sua simplicidade e facilidade de interpretação, tornando-se uma ferramenta muito importante na ciência de dados.

Este trabalho propõe uma aceleração da Predição Inter-Quadros do VVC, com foco no TZS e na AME, utilizando 24 Árvores de Decisão, duas para cada tamanho de CB, para reduzir a complexidade computacional e o tempo total de codificação, sem comprometer significativamente a eficiência de codificação.

## 2. METODOLOGIA

Os experimentos foram realizados no software de referência do VVC: *VVC Test Model* (VTM) versão 16.2 (SUEHRING, 2023), usando a configuração *Random-Access*, com foco específico na avaliação e aceleração das etapas do TZS e da AME. Os Parâmetros de Quantização, do inglês *Quantization Parameters* (QP), utilizados foram 22, 27, 32 e 37, conforme especificado pelas Condições Comuns de Teste, do inglês *Common Test Conditions* (CTC) (BOSSSEN et al., 2020). Todos os experimentos foram realizados em um servidor Intel Xeon CPU E5-2640 v3 @ 2,60 GHz, com oito núcleos e 32 GB de RAM.

A princípio, foi realizada uma análise de tempo de cada etapa do TZS e da AME. Então as *features* foram extraídas do VTM para o TZS, seguida pela implementação do Aprendizado de Máquina no VTM utilizando as *features* para otimizar o TZS. Em seguida, o processo foi repetido para a AME com a extração das suas *features* no VTM com o TZS acelerado, e por fim, foi implementado o novo Aprendizado de Máquina no VTM com as *features* da AME para otimizar a AME.

Considerando as etapas do TZS e os modos da AME apresentados anteriormente, seu consumo de tempo é analisado com base no processo de codificação dos vídeos selecionados. Para esta análise, foram codificados os primeiros 32 quadros de 23 sequências de vídeo diferentes, como definido pelas CTC. O VTM foi modificado apenas para inclusão de contadores de tempo. Estas sequências foram: três da classe A1 (*Tango2*, *FoodMarket4* e *Campfire*), três da classe A2 (*CatRobot*, *DaylightRoad2* e *ParkRunning3*), cinco de classe B (*MarketPlace*, *RitualDance*, *Cactus*, *BasketballDrive* e *BQTerrace*), quatro da classe C (*BasketballDrill*, *BQMall*, *PartyScene* e *RaceHorsesC*), quatro da classe D (*BasketballPass*, *BQSquare*, *BlowingBubbles* e *RaceHorses*) e quatro da classe F (*SlideEditing*, *SlideShow*, *BasketballDrillText* e *ArenaOfValor*). Os resultados médios desta análise estão apresentados na Figura 1.

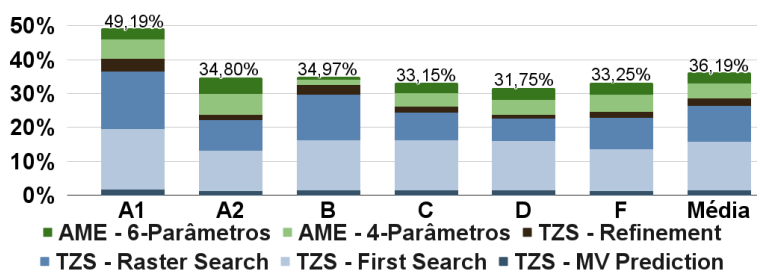


Figura 1: Tempo médio usado por cada etapa do TZS e AME em relação ao tempo total de codificação do VVC, considerando diferentes classes de vídeos.

A Figura 1 mostra que o TZS consome em média 28,52% do tempo total de codificação, sendo as etapas de *First Search* e *Raster Search* as mais demoradas. De modo similar, a AME representa em média 7,66% do tempo total de codificação.

A aceleração do VVC proposta neste trabalho usa modelos de Árvore de Decisão para pular seletivamente os três últimos passos do TZS e todo o processo de AME sob condições específicas, conforme ilustrado na Figura 2 com as decisões das DTs destacadas em azul e verde para o TZS e para a AME, respectivamente.

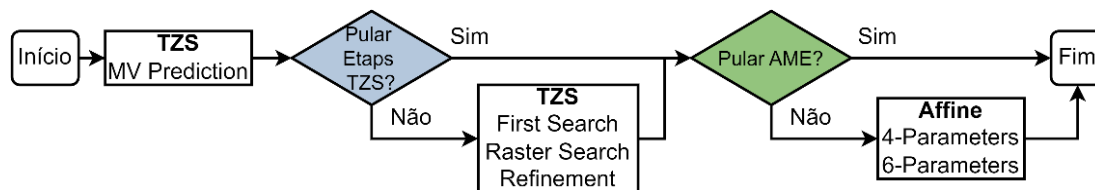


Figura 2: Fluxograma do método proposto.

Primeiro, o TZS foi otimizado extraíndo suas *features*, treinando 12 Árvores de Decisão e implementando-as no VTM. Em seguida, a AME foi otimizada em cima da otimização do TZS, utilizando o VTM otimizado para o TZS para aplicar os mesmos passos na AME: extração de *features*, treinamento e implementação das outras 12 Árvores de Decisão. Isso resultou em um VTM com ambas as otimizações implementadas.

Para extrair *features* relevantes para as etapas de codificação visadas para otimização, o codificador VTM foi executado normalmente com contagens adicionais de tempo de estágios específicos e a extração de dados adicionais. Foi utilizado um novo conjunto de 12 sequências de vídeo com diferentes resoluções: quatro vídeos em HD (*dark*, *KristenAndSara*, *Netflix\_DinnerScene* e *Netflix\_DrivingPOV*), dois vídeos em Full HD (*Netflix\_TunnelFlag* e *rush\_field\_cuts*) e seis vídeos em Ultra HD (*Beauty*, *BuildingHall2*, *Jockey*, *Lips*, *NetflixDancers* e *Netflix\_ToddlerFountain*). Os primeiros 16 quadros de cada sequência foram codificados para extrair as *features*, e cada sequência foi codificada quatro vezes, uma para cada QP. O processo de treinamento de cada Árvore de Decisão é mostrado na Figura 3.

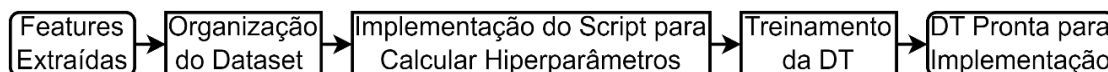


Figura 3: Fluxograma do processo para criação e treinamento de uma DT.

Os hiperparâmetros calculados e utilizados foram: *Criterion* (*entropy* ou *gini*), *Min Samples Split*, *Min Samples Leaf*, *Max Features*, *Max Depth* e *Max Leaf Nodes*.

### 3. RESULTADOS E DISCUSSÃO

Os primeiros 32 quadros das 23 sequências de vídeo usadas anteriormente para a análise de TZS e AME foram novamente codificados quatro vezes, uma vez para cada QP, mas desta vez usando o VTM 16.2 desenvolvido neste trabalho, com TZS e AME acelerados, com temporizadores adicionais para cronometrar cada etapa das ferramentas TZS e AME otimizadas com Árvores de Decisão. É importante realçar que as sequências das CTC utilizadas nestes experimentos são completamente diferentes daquelas usadas durante o processo de treinamento.

A avaliação dos resultados dos experimentos usou as métricas Redução de Tempo (TR), Redução de Tempo TZS (TR TZS) e Redução de Tempo AME, além da métrica *Bjontegaard Delta-BitRate* (BD-BR) para avaliar a eficiência de codificação. Os resultados médios para todas as sequências e todos os QPs estão apresentados na Tabela 1, organizados por Classe das CTC.

Tabela 1: Resultados da redução de tempo e eficiência de codificação.

Classe das Sequências de Vídeo	TR	TR TZS	TR AME	BD-BR
Média da Classe A1	18,45%	44,65%	52,19%	0,25%
Média da Classe A2	21,81%	56,51%	61,50%	1,24%
Média da Classe B	23,63%	69,81%	65,23%	1,25%
Média da Classe C	24,26%	70,72%	71,73%	0,77%
Média da Classe D	21,69%	72,60%	75,43%	1,02%
Média da Classe F	16,09%	58,62%	55,40%	0,85%
Média Geral	20,99%	62,15%	63,58%	0,90%

#### 4. CONCLUSÕES

Este trabalho apresentou uma solução baseada em Aprendizado de Máquina para reduzir o esforço computacional dos processos TZS e AME no VVC. A solução utiliza modelos de Árvores de Decisão para pular seletivamente três das quatro etapas do TZS e todo o processo da AME. No total, 24 Árvores de Decisão foram treinadas, duas para cada tamanho de CB suportados pelo TZS e AME no VVC.

O software VTM acelerado alcançou uma redução média de 20,99% no tempo total de codificação do VVC. Especificamente, reduziu o tempo médio de processamento do TZS em 62,15% e o tempo médio de processamento da AME em 63,58%, com uma perda média de eficiência correspondente de 0,90% em BD-BR. Este trabalho alcançou uma ótima redução no tempo de codificação, mantendo uma baixa perda de eficiência de codificação.

#### 5. REFERÊNCIAS BIBLIOGRÁFICAS

BOSSEN, F.; BOYCE, J.; SUEHRING, K.; LI, X.; VADIM, S. **VTM Common Test Conditions and Software Reference Configurations for SDR Video**. Document WG 05 MPEG Joint Video Coding Team(s) with ITU-T SG 16 JVET-T2010.ed., out. 2020. Online.

CISCO. **Cisco Annual Internet Report (2018–2023)**. Cisco Systems, Inc., San José, EUA, 09 mar. 2020. Especiais. Acessado em 08 fev. 2024. Online. <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>

LOOSE, M.; VIANA, R.; SAGRILO, F.; SANCHEZ, G.; CORRÊA, G.; AGOSTINI, L. A Hardware-Friendly and Configurable Heuristic Targeting VVC Inter-Frame Prediction. **IEEE INTERNATIONAL CONFERENCE ON ELECTRONICS, CIRCUITS AND SYSTEMS (ICECS)**, Glasgow, Reino Unido, 2022. **Anais**. . . 2022. p.1–4.

SUEHRING, K. **VTM-16.2**. JVET, 23 mai. 2022. Acessado em 08 fev. 2023. Online. Disponível em: [https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware\\_VTM/-/releases/VTM-16.2](https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/releases/VTM-16.2).

VIANA, R.; SAGRILO, F.; FERREIRA, R.; LOOSE, M.; PORTO, M.; CORRÊA, G.; AGOSTINI, L. A Hardware-Friendly Fast VVC Test Zone Search Algorithm Using Machine Learning. **IEEE 15TH LATIN AMERICA SYMPOSIUM ON CIRCUITS AND SYSTEMS (LASCAS)**, Punta del Este, Uruguai, 2024. **Anais**. . . 2024. p.1–5.