

Aprendizado de Máquina Aplicado à Redução do Custo Computacional do Test Zone Search e da Estimação de Movimento Affine do Codificador VVC

Ramiro Gomes da Silva Viana
rgsviana@inf.ufpel.edu.br
Universidade Federal de Pelotas (UFPel)
Pelotas, RS, Brazil

Luciano Volcan Agostini
agostini@inf.ufpel.edu.br
Universidade Federal de Pelotas (UFPel)
Pelotas, RS, Brazil

ABSTRACT

As the demand for video transmission surges in remote work, education, and streaming services, the need for continuous advancements in video encoding technologies becomes increasingly evident. Adapting to the evolving requirements of efficient video delivery and consumption, Versatile Video Coding (VVC) emerges as the state-of-the-art. This dissertation focuses on two key encoding tools within VVC Inter-Frame prediction: Test Zone Search (TZS) and Affine Motion Estimation (AME). A fast TZS and AME approach using Machine Learning is presented, specifically employing Decision Trees. The proposed approach achieved an average reduction of over 20% in total VVC encoding time while maintaining less than a 1% impact on BD-BR coding efficiency.

KEYWORDS

Versatile Video Coding, Test Zone Search, Estimação de Movimento Affine, Árvores de Decisão, Redução de Custo Computacional

1 INTRODUÇÃO

Atualmente, observa-se um aumento significativo na demanda por tecnologias que suportem o processamento de vídeos digitais. Este aumento é especialmente notável dada a contagem global de usuários de internet, que atingiu mais de 5 bilhões de pessoas, com aproximadamente 90% do tráfego da internet sendo atribuído a vídeos digitais [3]. Em função do elevado custo na transmissão e no armazenamento desses vídeos, o desenvolvimento de codificadores de vídeo eficientes tornou-se imperativo. Eles são projetados para comprimir os dados de forma expressiva, mantendo a qualidade da imagem. Por outro lado, a codificação de vídeo requer um enorme esforço computacional.

Nos últimos anos, o cenário da tecnologia de vídeo tem experimentado um crescimento substancial, com a popularização das resoluções 4k *Ultra High Definition* (UHD) e 8k *Full Ultra High Definition* (FHD). Consequentemente, vários grupos têm dedicado esforços para desenvolver novos codificadores de vídeo capazes de processar de forma eficiente esse tipo de mídia. Uma adição notável a esse conjunto de codificadores é o *Versatile Video Coding* (VVC), que foi oficialmente lançado em julho de 2020 e é o atual padrão de codificação de vídeo do estado da arte da ISO e ITU-T [2].

Apesar de suas exigências computacionais substanciais, o VVC se destaca por suas taxas de compressão muito elevadas, superando outros codificadores comerciais em eficiência, graças às suas ferramentas de codificação avançadas que permitem uma compressão de vídeo altamente eficiente. A predição Inter-Quadros do VVC incorpora ferramentas-chave para otimizar a codificação de vídeo e explorar a redundância espacial. Duas destas ferramentas são o *Test Zone Search* (TZS) e a Estimação de Movimento *Affine* (AME). Tanto o TZS quanto a AME suportam 12 tamanhos de Blocos de Codificação, do inglês *Coding Blocks*, entre 16×16 e 128×128 .

O *Test Zone Search* é conhecido por sua velocidade e eficiência no processo da Estimação de Movimento (ME). No entanto, mesmo sendo um algoritmo rápido de ME, o TZS é uma das ferramentas com maior custo computacional no padrão VVC. O TZS consiste em quatro etapas: Predição do Vetor de Movimento, Primeira Busca, Busca em *Raster* e Refinamento [4]. A etapa da Predição do Vetor de Movimento sempre precisa ser executada pelo codificador.

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 e o modelo de 6-Parâmetros para movimentos mais complexos [2].

Uma das abordagens atuais com maior potencial para reduzir o custo computacional dos codificadores de vídeo é o Aprendizado de Máquina. Diversos trabalhos na literatura apresentam soluções para diferentes codificadores e suas respectivas ferramentas, utilizando técnicas variadas de Aprendizado de Máquina, como a Árvore de Decisão, do inglês *Decision Tree* (DT). A Árvore de Decisão se destaca pela sua simplicidade, facilidade de interpretação e implementação direta, tornando-se uma ferramenta crucial na ciência de dados.

2 ACELERAÇÃO DA PREDIÇÃO INTER-QUADROS DO VVC

Esta dissertação de mestrado apresenta o desenvolvimento de modelos de Aprendizado de Máquina com o objetivo de reduzir o custo computacional do TZS e do AME com impactos mínimos na eficiência de codificação. Para isso, foram treinados 24 modelos de Árvore de Decisão, 12 para pular as três etapas finais do TZS e 12 para pular todo o processo do AME. Um modelo DT foi treinado para cada um dos 12 tamanhos de bloco suportados tanto pelo TZS quanto pelo AME. A Figura 1 apresenta a solução rápida desenvolvida para Predição Inter-Quadros no VVC, com as decisões baseadas nas DTs

destacadas em azul e verde. Cada decisão na Figura 1 representa um grupo de 12 modelos de DTs, um para cada tamanho de bloco.

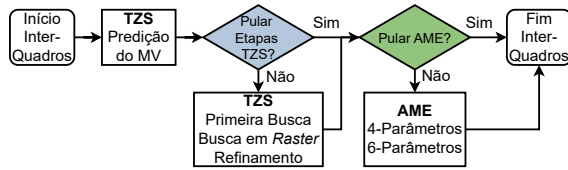


Figura 1: Fluxograma da solução proposta para acelerar o TZS e a AME utilizando Árvores de Decisão.

Todo o processo de organização de *datasets* e treinamentos e testes das Árvores de Decisão foi feito em Python com a biblioteca *Scikit-Learn*, através da plataforma *Google Colab*.

A extração de *features* utilizou 12 sequências com diferentes resoluções, oriundas dos *datasets* UVG, NETVC e JVET [6]. O *software* de referência *VVC Test Model* (VTM) versão 16.2 [5] foi modificado para permitir a captura das *features*. Foram codificados os 16 primeiros quadros usando quatro valores do Parâmetro de Quantização, do inglês *Quantization Parameters* (QP), (22, 27, 32 e 37), conforme definido pelas *Common Test Conditions* (CTCs) [1].

As *features* para treinamento das DTs para o TZS e o AME foram selecionadas por sua relevância para o processo de estimação de movimento, em um total de 23 *features* usadas nos modelos do TZS e 37 nos modelos do AME.

Os dados foram organizados em 24 *datasets* balanceados, com 100.000 linhas em cada um. A seguir foram definidos os hiperparâmetros de cada DT através do algoritmo *Random Search*. Então foi realizado o treinamento das 24 DTs. Para teste, foram usadas outras três sequências, também oriundas dos *datasets* UVG, NETVC e JVET. Os modelos foram avaliados separadamente, alcançando F1-Scores entre 0,877 e 0,954. Esses modelos foram traduzidos para C++ e inseridos no VTM para as avaliações em um cenário real.

3 RESULTADOS EXPERIMENTAIS

O VTM na versão 16.2 foi usado na avaliação dos modelos treinados em um cenário real. Nesse caso, as especificações das CTCs na configuração *Random Access* foram integralmente seguidas. Então foram usados quatro QPs (22, 27, 32 e 37) para codificar as 23 sequências de vídeo definidas pelas CTCs (diferentes das usadas no treinamento). Essas sequências estão categorizadas nas classes A1, A2, B, C, D e F. Todos os experimentos foram realizados em um servidor equipado com um processador Intel Xeon CPU E5-2640 v3 @ 2,60 GHz, com oito núcleos e 32 GB de memória RAM.

Os ganhos no tempo de codificação foram medidos utilizando três métricas: Redução de Tempo (RT) para ganhos gerais no codificador; Redução de Tempo do TZS (RT_{TZS}) para os ganhos na etapa do TZS; e Redução de Tempo da AME (RT_{AME}) para os ganhos na etapa da AME. Já as perdas na eficiência de codificação foram avaliadas por meio do Bjøntegaard Delta-Bitrate (BD-BR), uma métrica definida pelas CTCs e que indica, para uma mesma qualidade objetiva, a variação percentual na quantidade de bits necessária para representar os vídeos usando um codificador modificado em comparação com o codificador de referência. Os resultados para todas as sequências de vídeo são apresentados na Tabela 1.

Tabela 1: Resultados da implementação proposta.

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

Com os resultados mostrados na Tabela 1, é possível observar que o VTM acelerado desenvolvido nesta dissertação alcançou uma redução média de 20,99% no tempo total de codificação do VVC. Especificamente, reduziu o tempo de processamento do TZS em 62,15% e o tempo de processamento do AME em 63,58%. Além disso, foi observado um leve decréscimo na eficiência de compressão, com um aumento médio de 0,90% no BD-BR. Além disso, o impacto das execuções das Árvores de Decisão no VTM foi mínimo, resultando em custo de apenas 0,30% no tempo total de codificação.

4 CONCLUSÕES

Esta dissertação de mestrado apresentou uma solução rápida para Predição Inter-Quadros no VVC baseada em Aprendizado de Máquina, com o objetivo de reduzir o esforço computacional das ferramentas TZS e AME do VVC. A solução utiliza modelos de Árvore de Decisão para pular seletivamente três das quatro etapas do TZS e todo o processo do AME. No total, foram treinadas 24 Árvores de Decisão, uma para cada um dos 12 tamanhos de bloco suportados pelo TZS e pelo AME no VVC.

O *software* VTM acelerado alcançou uma redução média de 20,99% no tempo total de codificação do VVC, com uma perda média de eficiência de codificação em BD-BR de somente 0,90%. Desta forma, foi possível alcançar uma redução de custo computacional expressiva, com uma perda muito pequena na eficiência de codificação.

ACKNOWLEDGMENTS

Este trabalho foi financiado por CAPES, CNPq e FAPERGS.

REFERENCES

- [1] Frank Bossen, Jill Boyce, Xiang Li, Vadim Seregin, and Karsten Suhling. 2020. *VTM Common Test Conditions and Software Reference Configurations for SDR Video*. https://jvet-experts.org/doc_end_user/current_document.php?id=10545
- [2] Benjamin Bross, Jianle Chen, Jens-Rainer Ohm, Gary J. Sullivan, and Ye-Kui Wang. 2021. Developments in International Video Coding Standardization After AVC, With an Overview of Versatile Video Coding (VVC). *Proc. IEEE* 109, 9 (2021), 1463–1493. <https://doi.org/10.1109/JPROC.2020.3043399>
- [3] CISCO. 2023. *Cisco Annual Internet Report (2018–2023)*. <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>
- [4] Rafael Ferreira, Luciano Agostini, Cláudio M. Diniz, and Bruno Zatt. 2023. Evaluation of Imprecise Subtractors into Test Zone Search for VVC Encoding. In *2023 36th SBC/SBMicro/IEEE/ACM Symposium on Integrated Circuits and Systems Design (SBCCI)*. 1–6. <https://doi.org/10.1109/SBCCI60457.2023.10261956>
- [5] Karsten Suehring. 2022. *VTM-16.2*. https://vcgit.hhi.fraunhofer.de/jvet/VVCSofware_VTM/-/releases/VTM-16.2
- [6] Ramiro Gomes da Silva Viana. 2025. *Acelerando o Algoritmo de Estimação de Movimento Affine do Padrão VVC Utilizando Árvores de Decisão*. Dissertação de Mestrado. Universidade Federal de Pelotas (UFPEL), Pelotas, RS, Brasil. <https://guaiaca.ufpel.edu.br/handle/prefix/15988> Programa de Pós-Graduação em Computação (PPGC).