

## VALIDAÇÃO DE UM ESQUEMA BASEADO EM DATA MINING PARA REDUÇÃO DE COMPLEXIDADE COMPUTACIONAL EM CODIFICADORES DE VÍDEO DE ALTA EFICIÊNCIA

DOUGLAS CORRÊA; GIOVANNI AVILA; IAGO STORCH; LUCIANO AGOSTINI;  
GUILHERME CORRÊA

*Universidade Federal de Pelotas – Grupo de Arquiteturas e Circuitos Integrados*  
{dcorrea, gdadavila, icstorch, agostini, gcorrea}@inf.ufpel.edu.br

### 1. INTRODUÇÃO

O mais novo padrão de codificação de vídeo, o *High Efficiency Video Coding* (HEVC) (JCT-VC, 2013), foi lançado em 2013 e desde então vem gradualmente substituindo o seu antecessor, o padrão H.264/AVC, devido à sua grande eficiência de compressão. Contudo, dependendo da resolução dos vídeos e da configuração do codificador utilizada, a complexidade computacional do codificador HEVC pode ser de 10% a 500% maior que aquela do codificador H.264/AVC, o que representa um problema especialmente grave em dispositivos móveis, com disponibilidade limitada de energia, e em sistemas de tempo real, que requerem um fluxo contínuo de codificação e transmissão (CORRÊA, 2012).

O trabalho de CORRÊA et al. (2014), que serve de base para a investigação apresentada neste resumo, utiliza uma estratégia baseada em *Data Mining* para construir árvores de decisão de baixa complexidade que substituem o processo exaustivo e extremamente custoso de decisão de algumas estruturas de particionamento existentes no padrão HEVC, nomeadamente as *Coding Units* (CUs). A estratégia apresentou uma redução média de 37% na complexidade computacional do codificador HEVC, com um custo de apenas 0.28% em termos de Bjøntegaard Delta (BD)-rate (BJONTEGAARD, 2001), uma métrica de eficiência de compressão comumente utilizada na área.

Em (CORRÊA, 2014), o processo de treinamento das árvores foi realizado de forma *off-line*, a partir da codificação de um conjunto de sequências de vídeo com o codificador HEVC, coleta de dados e aplicação de um algoritmo de aprendizagem de máquina sobre os dados coletados. As árvores obtidas foram implementadas no codificador HEVC e o seu desempenho foi finalmente testado através da codificação de um segundo conjunto de sequências de vídeo totalmente diferente daquele utilizado no treinamento. Este processo permitiu verificar a grande eficiência das árvores obtidas no treinamento *off-line*, mas não garante a eficiência do método baseado em *Data Mining* para qualquer caso, visto que apenas um treinamento com apenas um conjunto de dados foi realizado. Por exemplo, qual seria o impacto, tanto em termos de complexidade computacional quanto em termos de eficiência de compressão, se as árvores de decisão fossem treinadas com um conjunto completamente diferente de vídeos? Além disso, o quanto robustas são as árvores obtidas e o quanto precisos são os seus testes?

Essas foram as questões motivadoras para a investigação apresentada neste resumo, que consiste numa validação do esquema de redução de complexidade computacional apresentado em (CORRÊA, 2014) e numa análise da robustez das árvores de decisão treinadas. A metodologia utilizada para a validação do esquema e para a análise de robustez é apresentada na seção 2. A seção 3 apresenta os resultados com base nos testes realizados e a seção 4 apresenta as conclusões obtidas sobre a análise de resultados.

## 2. METODOLOGIA

A utilização de técnicas de *Data Mining* permite determinar o valor de variáveis dependentes através da observação dos valores de alguns atributos em um conjunto de dados, identificando regularidades e construindo regras de generalização que são representadas sob a forma de modelos. Devido à sua baixa complexidade computacional e grande facilidade de compreensão, os modelos baseados em árvores de decisão foram escolhidos para utilização em (CORRÊA, 2014). Tanto em (CORRÊA, 2014) quanto neste trabalho, o processo de *Data Mining* foi realizado com o auxílio da plataforma WEKA. Os arquivos de entrada do WEKA são do formato *Attribute-Relation File Format* (ARFF), como mostrado na Figura 1(a). O arquivo consiste de um cabeçalho com a lista de atributos e de uma seção com os dados utilizados para treinamento, os quais foram coletados através da inserção de sub-rotinas no software de referência do codificador HEVC, o *HEVC Test Model* (HM) (JCT-VC, 2013). Com base nos dados coletados, o algoritmo C4.5 implementado na plataforma WEKA foi utilizado para a geração das árvores de decisão que realizam o particionamento das CUs. A Figura 1(b) apresenta um exemplo de árvore gerada, onde uma folha igual a '0' representa a decisão de não sub-particionar a CU e uma folha igual a '1' representa a decisão de sub-particionar a CU em quatro CUs menores. O treinamento e a validação das árvores de decisão em (CORRÊA, 2014) foram realizados com os vídeos do conjunto A e B da Tabela 1, respectivamente.

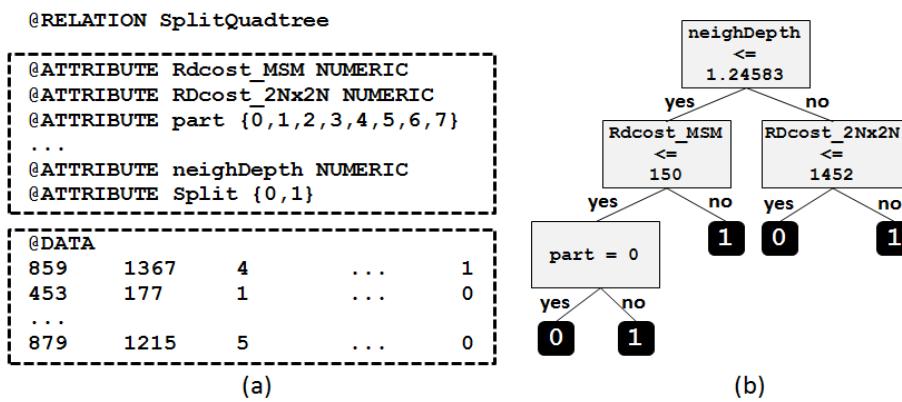


Figura 1: (a) Exemplo de arquivo ARFF; (b) Exemplo de árvore de decisão.

### 2.1. Validação do Esquema Proposto

A primeira parte desta pesquisa foi a validação do esquema proposto por CORRÊA et al. (2014). Para isso, o mesmo procedimento descrito nos parágrafos anteriores foi executado, com algumas particularidades.

Primeiramente, foram recolhidos os dados para treinamento das árvores através da execução do codificador HM modificado, com sub-rotinas que permitem a extração de resultados intermediários da codificação. Porém, ao contrário de (CORRÊA, 2014), foram utilizados os vídeos do conjunto B para treinamento, ao invés dos vídeos do conjunto A. Os dados foram coletados a partir da codificação de todos os dez vídeos do conjunto B para quatro QPs diferentes (22, 27, 32 e 37), totalizando 40 codificações. Os dados coletados para cada combinação de vídeo e QP foram organizados conforme o formato ARFF e um arquivo de treinamento foi criado para cada possível tamanho de CU que permite sub-particionamento: 16x16, 32x32 e 64x64.

Tabela 1: Vídeos utilizados no treinamento e na validação das árvores de decisão.

<b>Conjunto A</b>		<b>Conjunto B</b>	
<b>Nome</b>	<b>Resolução</b>	<b>Nome</b>	<b>Resolução</b>
<i>BlowingBubbles</i>	416x240	<i>BasketballPass</i>	416x240
<i>RaceHorses</i>	416x240	<i>BQSquare</i>	416x240
<i>PartyScene</i>	832x480	<i>BasketballDrill</i>	832x480
<i>BQMall</i>	832x480	<i>ChinaSpeed</i>	1024x768
<i>SlideShow</i>	1280x720	<i>Kimono1</i>	1024x768
<i>vidyo1</i>	1280x720	<i>SlideEditing</i>	1280x720
<i>BasketballDrive</i>	1920x1080	<i>BQTerrace</i>	1920x1080
<i>ParkScene</i>	1920x1080	<i>Cactus</i>	1920x1080
<i>NebutaFestival</i>	1920x1080	<i>PeopleOnStreet</i>	1920x1080
<i>Traffic</i>	1920x1080	<i>SteamLocomotive</i>	1920x1080

A partir dos três arquivos ARFF, o software WEKA foi utilizado para o treinamento das três árvores de decisão com o algoritmo *C4.5* (implementação *J48*). As árvores treinadas foram implementadas em C++ no codificador HM e posteriormente foram realizados os testes de validação com o conjunto A de vídeos e os QPs 22, 27, 32 e 37, conforme relata a seção 3 deste resumo.

## 2.2. Análise da Robustez das Árvores de Decisão

A segunda parte do trabalho consistiu em analisar a robustez das árvores geradas por CORRÊA et al. (2014) através da redução da precisão dos *thresholds* utilizados nos testes para definição do sub-particionamento das CUs.

Na primeira análise realizada, todos os *thresholds* foram incrementados em 5%. Para isso, o código de cada uma das três árvores de decisão (16x16, 32x32 e 64x64) foi alterado diretamente no codificador HM. Tomando-se a árvore ilustrada na Figura 1(b) como exemplo, os *thresholds* 1.24583, 150 e 1452 passaram a ser 1.30812, 157.5 e 1524.6, respectivamente. Na segunda análise realizada, os mesmos passos foram tomados, com os *thresholds* decrementados em 5% ao invés de incrementados. Nas duas análises, as duas novas versões do codificador HM foram utilizadas para codificar os vídeos do conjunto A (Tabela 1) com QPs 22, 27, 32 e 37, conforme relata a seção 3 deste resumo.

## 3. RESULTADOS E DISCUSSÃO

Para fins de comparação, a Tabela 3 reproduz os resultados apresentados em (CORRÊA, 2014) e a Tabela 4 apresenta os resultados obtidos com as árvores treinadas neste trabalho. Como se pode perceber ao comparar as duas tabelas, ambos conjuntos de árvores são extremamente similares do ponto de vista topológico, com pequenas variações na sua profundidade, número de nodos de teste e folhas. Além disso, as árvores treinadas apresentaram variações de desprezíveis em relação às apresentadas em (CORRÊA, 2014) em termos de precisão da decisão. No pior caso, notou-se uma diferença de apenas 1.3% na precisão entre as árvores treinadas com os dois diferentes conjuntos de vídeos.

A Tabela 5 mostra os resultados em termos de variação da taxa de bits (BD-rate), qualidade da imagem (BD-PSNR) e redução da complexidade computacional (RCC) para todos os experimentos realizados. A primeira linha de resultados reproduz aqueles apresentados em (CORRÊA, 2014), para fins de comparação. Novamente, nenhum dos experimentos realizados nesta pesquisa

apresentou resultados expressivamente diferentes daqueles apresentados no trabalho de referência. No pior caso, notou-se uma diferença de 0.09% em BD-rate, 0.02 dB em BD-PSNR e 5% em RCC. Estes resultados permitem, portanto, validar a estratégia de redução de complexidade baseada em técnicas de *Data Mining* e confirmam a robustez das árvores apresentadas em (CORRÊA, 2014).

Tabela 3: Características das árvores treinadas com o conjunto A.

Tamanho da CU	Profundidade	Nodos de testes	Folhas	Precisão
64x64	5	6	19	84.2%
32x32	8	20	33	84.5%
16x16	9	23	44	84.6%

Tabela 4: Características das árvores treinadas com o conjunto B.

Tamanho da CU	Profundidade	Nodos de testes	Folhas	Precisão
64x64	7	9	22	82.9%
32x32	8	20	33	83.6%
16x16	9	26	45	84.6%

Tabela 5: Resultados da análise de robustez das árvores.

Experimento	BD-rate (%)	BD-PSNR (dB)	RCC (%)
Treinamento Conj. A	+0.28	-0.01	37
Treinamento Conj. B	+0.37	-0.02	32
Thresholds +5%	+0.32	-0.01	35
Thresholds -5%	+0.21	-0.01	34

#### 4. CONCLUSÕES

Este trabalho apresentou os resultados de uma investigação que teve como finalidade validar a estratégia para redução de complexidade computacional do codificador HEVC proposta por CORRÊA et al. (2014) e analisar a robustez das árvores de decisão apresentadas naquele trabalho. Após um novo ciclo de treinamento, implementação e teste das novas árvores de decisão obtidas com um segundo conjunto de sequências de vídeo, concluiu-se que os resultados apresentados naquele trabalho não estão fortemente vinculados ao conjunto de sequência de vídeos utilizado para treinamento. Além disso, a análise de robustez permitiu verificar que pequenas variações ( $\pm 5\%$ ) nos *thresholds* das árvores de decisão implementadas não geraram resultados significativamente diferentes em termos de eficiência de compressão e redução de complexidade.

#### 5. REFERÊNCIAS BIBLIOGRÁFICAS

- JCT-VC. High Efficiency Video Coding text specification draft 10, doc. JCTVC-L1003, Genebra, Suíça, 2013.
- CORRÊA, G.; et al. Performance and Computational Complexity Assessment of High Efficiency Video Encoders. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 22, n. 12, p. 1899-1909, 2012.
- CORRÊA, G.; et al. Classification-based early termination for coding tree structure decision in HEVC. In: **21st IEEE International Conference on Electronics, Circuits and Systems**, Marselha, França, p.239-242, 2014
- BJONTEGAARD, G.. Calculation of average PSNR differences between RD-curves, doc. VCEG-M33, Austin, Texas, 2001.