

Codificação Distribuída de Vídeo Digital Utilizando Códigos Turbo e Wavelets de Segunda Geração

André F. Ponchet, Yuzo Iano.

Abstract — Distributed video coding is a new paradigm for video compression in opposition over the existing video coding standards like MPEG-x and H.26x families. The present work proposes a video compression scheme based on the lossy distributed source coding theory, performing the encoding process of the video signal without motion estimation algorithms. The decoding process is performed in a iterative manner and explores the statistical dependency of the video frames of the original sequence. This approach gives a great encoding runtime reduction. Simulation results show the good performance of the proposed codec in comparison with the state of the art video compression standard, H.264/AVC.

Key words — Distributed coding, video compression, turbo codes, second generation wavelets.

Resumo — A codificação distribuída de vídeo constitui um novo paradigma em compressão de vídeo frente aos codificadores híbridos da família MPEG-x e H.26x. Neste trabalho é apresentado um *codec* de vídeo baseado na teoria da codificação distribuída com perdas, onde a codificação da seqüência de vídeo não emprega estimação de movimento. O processo de decodificação é feito de forma iterativa e explora a dependência estatística entre os quadros da seqüência de vídeo original. Esta abordagem permite uma redução bastante significativa no tempo de processamento envolvido na compressão do sinal de vídeo. Os resultados obtidos em simulações comprovam o bom desempenho do *codec* proposto em relação ao padrão estado da arte em compressão de vídeo, o H.264/AVC.

Palavras chave — Codificação distribuída, compressão de vídeo, códigos turbo, *wavelets* de segunda geração.

I. INTRODUÇÃO

Nos últimos anos, a demanda por serviços de multimídia vem se tornando cada vez mais crescente exigindo codificadores de vídeo cada vez mais eficientes. A contínua evolução nas pesquisas em compressão de vídeo digital se deve, basicamente a razões comerciais e técnicas. O fator comercial vem da necessidade de substituir a antiga tecnologia analógica por produtos que suportassem uma tecnologia mais eficiente do ponto de vista de armazenamento e transmissão.

Manuscrito recebido em 28 de novembro de 2005; revisado em 17 de Fevereiro de 2006.

A. F. Ponchet (ponchet@decom.fee.unicamp.br) e Y. Iano (yuzo@decom.fee.unicamp.br) pertencem ao Departamento de Comunicações da Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas – FEEC/UNICAMP. Av. Albert Einstein, 400 – Campinas - SP - Brasil - 13083-852.

O fator técnico advém das pesquisas envolvidas nos últimos anos em tecnologia de compressão de imagem e vídeo digitais. A combinação destes dois fatores permitiu o avanço na tecnologia de compressão de vídeo, nas mais diversas aplicações, como por exemplo: vídeo conferências, transmissão de vídeo digital em redes *wireless* e a televisão de alta definição, HDTV.

Em particular, sinais de vídeo digital necessitam de grandes fatores de compressão por parte do codificador em virtude da grande quantidade de informação associada. Para este fim, há uma demanda computacional intensa, sendo a redução do custo computacional o principal objeto de pesquisa dos codificadores propostos na literatura [1].

Basicamente, os codificadores de vídeo exploram a redundância espacial (entre *pixels* adjacentes) e a redundância temporal (inter quadros) do sinal original. A redundância espacial é explorada através do uso de transformadas e de quantizadores. As transformadas mais conhecidas na literatura são a DCT e a TWD. O uso de quantizadores leva a perda de informação do sinal original, mas ao mesmo tempo oferece fatores de compressão superiores aos codificadores que empregam técnicas de compressão sem perdas. A redundância temporal é explorada através de técnicas preditivas, onde cada quadro é predito através de algoritmos de estimação e compensação de movimento [1]. Na literatura, os codificadores que empregam técnicas por transformada e estimação e compensação de movimento, são ditos codificadores híbridos [2]-[3].

Neste trabalho, aborda-se a compressão distribuída de fonte, um novo paradigma de codificação, aplicado em compressão de sinais de vídeo [4]-[5]. O objetivo principal do trabalho é utilizar esta nova técnica de compressão, aliada a transformada *wavelet* discreta e códigos turbo, para implementar um *codec* de vídeo para aplicações em dispositivos móveis. A principal vantagem desse esquema de codificação, em relação aos codificadores da família MPEG é a baixa complexidade do codificador, tendo em vista que os quadros das seqüências de vídeo são intra codificados, ou seja, de forma independente sem o uso de algoritmos de estimação e compensação de movimento no codificador [5]. Portanto, a baixa complexidade do codificador viabiliza a sua implementação em dispositivos portáteis, como telefones celulares, tendo em vista que estes dispositivos possuem recursos de processamento bastante limitados. A pesquisa realizada na literatura disponível revelou que há poucos trabalhos que tratam o problema da codificação distribuída em aplicações de vídeo. Nos artigos pesquisados foram encontrados apenas trabalhos que utilizam esta técnica

combinada com o uso da transformada DCT e quantizadores lineares, revelando, portanto, que a pesquisa neste campo está ainda no seu início.

II. FUNDAMENTOS TEÓRICOS

A. Codificação distribuída sem perdas de fontes correlacionadas

A codificação distribuída de uma fonte de informação teve como origem o teorema proposto por *Slepian-Wolf* [6] para o caso da codificação sem perdas, e no teorema *Wyner-Ziv* [7]-[9], para o caso da codificação com perdas. Embora a teoria proposta nesses teoremas data da década de 70, somente nos últimos anos essa teoria tem tido aplicações em compressão de imagens e vídeo [10]-[12]. Em compressão de vídeo a teoria da codificação distribuída constitui um novo paradigma em relação aos algoritmos de compressão disponíveis na literatura [4]. O *codec* de vídeo que emprega esse novo conceito de compressão não faz uso de técnicas de estimação e compensação de movimento no codificador. Os quadros são codificados de forma independente, ou seja, intracodificados, utilizando dois esquemas de codificação diferentes [4]. A decodificação é feita de forma conjunta, a fim de explorar a correlação entre os quadros da seqüência de vídeo. Esta abordagem reduz consideravelmente a complexidade do codificador, permitindo o seu uso em dispositivos de reduzida capacidade de processamento e memória, como por exemplo, telefones celulares com câmeras digitais e *palm tops*.

O princípio da codificação distribuída de uma fonte de informação está ilustrado na Fig. 1, onde duas seqüências correlacionadas aleatórias geradas por uma fonte de informação são codificadas separadamente e decodificadas de forma conjunta:



Fig. 1. Codificação distribuída de duas fontes estatisticamente dependentes.

Neste algoritmo, dois codificadores, não necessariamente iguais, codificam as fontes de informação \mathbf{X} e \mathbf{Y} separadamente, com taxas R_x e R_y respectivamente, e enviam as seqüências de símbolos codificados para um único decodificador. A codificação distribuída sem perdas está fundamentada no teorema de *Slepian-Wolf* [6] cujo enunciado é apresentado de acordo com o teorema 1:

Teorema 1 (Teorema de Slepian-Wolf): Seja ε um número real arbitrário maior que zero. Sejam \mathbf{X} e \mathbf{Y} duas fontes de informação estatisticamente dependentes e R_x e R_y suas taxas de codificação associadas que satisfazem as equações 1, 2 e 3:

$$R_x > H(X/Y), \tag{1}$$

$$R_y > H(Y/X), \tag{2}$$

$$R_x + R_y > H(X,Y). \tag{3}$$

Então, para um código de bloco de comprimento n suficientemente grande, existe um código de bloco C composto pelos códigos C_x com taxa R_x e C_y com taxa R_y , tal que a probabilidade de erro p_e é menor que ε .

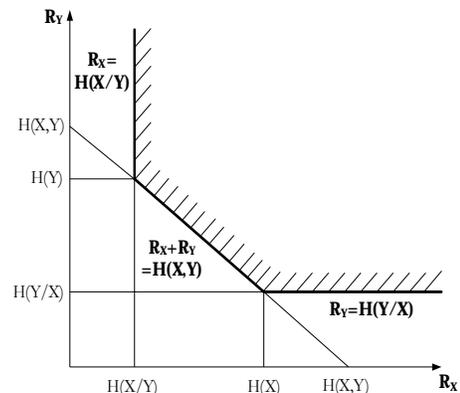


Fig. 2. Região das taxas admissíveis definida pelo teorema 1.

Em outras palavras, o teorema 1 garante que existe um código de bloco C de comprimento n tal que a probabilidade de erro na decodificação de um bloco desse código é tão pequena quanto se queira. O teorema 1 define a região das taxas admissíveis destacada na Fig. 2. Esta região é definida como sendo o conjunto convexo de todas as taxas R_x e R_y que satisfazem as equações 1, 2 e 3 [6].

B. Codificação distribuída com perdas de fontes correlacionadas

Na seção anterior, foi abordado o problema da codificação distribuída sem perdas, com base no teorema de *Slepian-Wolf*. Esta abordagem é adequada para codificação de fontes em que a reprodução exata da seqüência original é fundamental. Em compressão de imagem e vídeo é necessário atingir razões de compressão mais elevadas, em virtude das altas taxas de *bits* envolvidas. Desta forma, não é possível fazer uso de técnicas de compressão sem perdas em virtude das baixas razões de compressão atingidas por essas técnicas [13]. Algoritmos de compressão com perdas fornecem razões de compressão mais adequadas para esse tipo de aplicação e estabelecem um compromisso entre a taxa obtida e a distorção introduzida no processo de codificação [13]. O teorema de *Wyner-Ziv* estabelece a taxa necessária para codificar a seqüência \mathbf{X} do exemplo apresentado na Fig.1, sabendo-se que apenas o decodificador tem acesso à seqüência \mathbf{Y} e que a distorção média entre a seqüência original e a seqüência recuperada \mathbf{X}^* está sujeita à restrição dada pela equação 4:

$$E\{d(\mathbf{X}, \mathbf{X}^*)\} \leq D. \tag{4}$$

O teorema de *Wyner-Ziv* pode ser visto como uma generalização do caso descrito pelo teorema de *Slepian-Wolf*. Denominando $R^*(D)$ a taxa mínima para o sistema da Fig. 1,

o teorema de *Slepian-Wolf* determina que, para distorção d igual a zero, a equação 5 é válida:

$$R^*(0) = R_{X/Y}(0) = H(X/Y). \tag{5}$$

A teoria desenvolvida por *Wyner-Ziv* permite a dedução da função taxa-distorção $R^*(D)$ para o caso geral em que $D \geq 0$. Para $D \geq 0$, foi demonstrado em [9] que, de forma geral, a equação 6 é satisfeita:

$$R_{X/Y}^{WZ}(D) = R^*(D) \geq R_{X/Y}(D). \tag{6}$$

A figura 3 apresenta o esquema do codificador *Wyner-Ziv*:

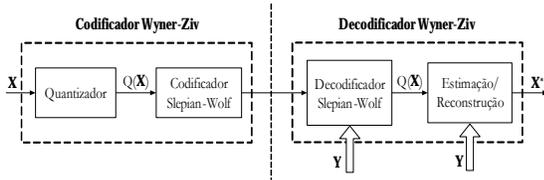


Fig. 3. Codificador *Wyner-Ziv*.

C. Compressão de vídeo baseada na teoria da codificação distribuída

Os padrões de compressão de vídeo da família MPEG-x e H.26x são baseados em uma arquitetura onde a codificação emprega uma transformação linear e um algoritmo de estimação e compensação de movimento. A teoria da codificação distribuída de fontes correlacionadas apresentada nas seções anteriores permite uma nova abordagem no processo de codificação do sinal de vídeo. O primeiro codificador implementado, a título de investigação, está apresentado na figura 4, de acordo com [4], [11]:

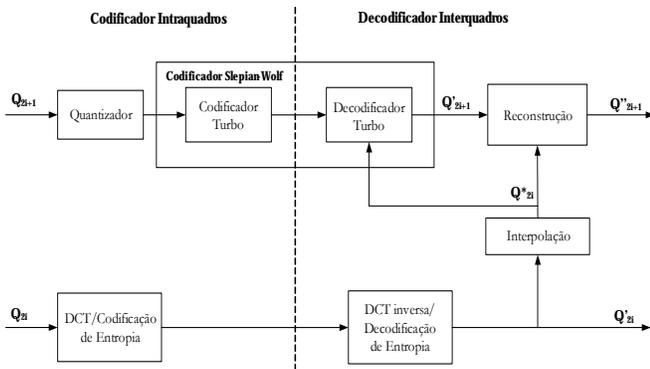


Fig. 4. Codificação distribuída de vídeo.

No codificador da Fig. 4 os quadros das seqüências de vídeo são divididos em dois grupos: o grupo formado pelos quadros ímpares, denominados de quadros *Wyner-Ziv* [4] e os quadros pares, denominado de quadros “chave” (do inglês, *key frames*). O processo de codificação dos quadros pares segue o mesmo padrão de um codificador de imagens estáticas, onde é utilizada a transformada DCT com blocos 8×8 pixels, seguida

de um quantizador linear e um codificador de entropia, composto por um algoritmo de corrida de zeros (do inglês, *run length coding*) e um codificador aritmético. A taxa de codificação foi variada através do passo de quantização e através do perfurador aplicado aos bits de paridade. O código turbo utilizado emprega dois codificadores convolucionais de taxa 4/5, idênticos aos codificadores utilizados em [4]. Os bits sistemáticos são descartados e apenas os bits de paridade gerados pelos codificadores convolucionais são enviados ao decodificador. A taxa de codificação é reduzida também através do puncionamento (do inglês, *puncturing*) dos bits de paridade. Portanto, o decodificador *Wyner-Ziv* dispõe dos quadros pares e dos bits de paridade para recuperar os quadros ímpares da seqüência de entrada. Os quadros pares recuperados no decodificador são a informação adicional (do inglês, *side information*) que o decodificador turbo dispõe além dos bits de paridade. A proposta deste trabalho consiste basicamente na modificação do codificador apresentado na Fig. 4, conforme o esquema apresentado na Fig. 5:

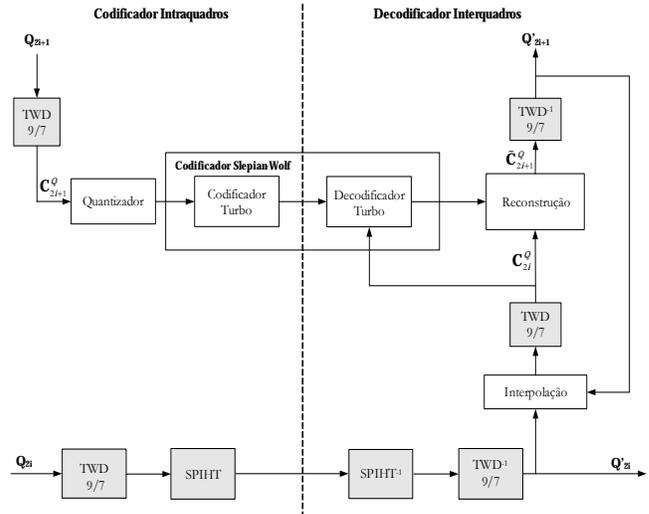


Fig. 5. Codificador de vídeo proposto usando códigos turbo e a TWD 9/7.

A transformada *wavelet* biortogonal 9/7 foi implementada pela técnica proposta em [15], [18]. Este algoritmo recebe o nome de *lifting* e foi utilizado na implementação do ISO JPEG 2000 [21], [22]. A motivação original para o desenvolvimento do algoritmo *lifting* era construir *wavelets* de “segunda geração” (do inglês, *second generation wavelet transform*), ou seja, *wavelets* obtidas sem a translação e dilatação de uma função, conforme a teoria apresentada originalmente por [23]-[24]. A implementação prática da TWD é feita através do banco de FIR, de acordo com a figura 6:

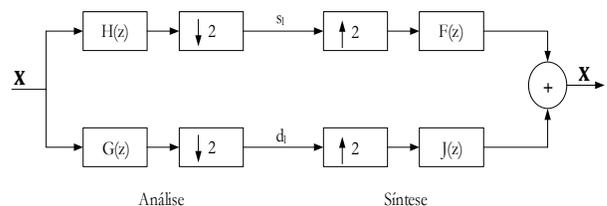


Fig. 6. Banco de filtros que implementam a TWD.

O algoritmo *lifting* é uma representação polifásica mais eficiente que o banco de filtros apresentado na figura 6. Na implementação da TWD 9/7, o esforço computacional é reduzido em 64% em relação ao algoritmo tradicional, de acordo com a tabela I [25]:

TABELA I

COMPARAÇÃO ENTRE O NÚMERO TOTAL DE OPERAÇÕES ENVOLVIDAS NO CÁLCULO DE ALGUNS FILTROS QUE IMPLEMENTAM A TWD.

TRANSFORMADA	BANCO DE FILTROS	LIFTING	GANHO DE DESEMPENHO
Haar	3	3	0%
Daubechies 4	14	9	56%
Daubechies 6	22	14	57%
9/7	23	14	64%
4/2 B-spline	17	10	70%

A estrutura polifásica do algoritmo *lifting* para o cálculo da transformada *wavelet* biortogonal 9/7 é apresentada na figura 7:

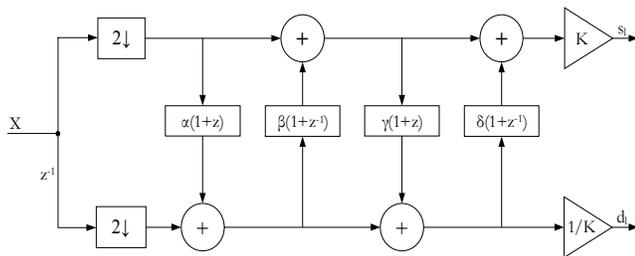


Fig. 7: Estrutura polifásica do algoritmo *lifting* para o cálculo da TWD biortogonal 9/7.

Os valores das constantes α , β , γ , e δ calculados pelo algoritmo descrito em [26]-[27] estão indicados na tabela 2:

Tabela 2: Constantes associadas ao cálculo da TWD 9/7 pelo algoritmo *lifting*.

α	-3/2
β	-1/16
γ	4/5
δ	15/16
K	$4\sqrt{2}/5$

III. RESULTADOS DE SIMULAÇÃO

Nesta seção são apresentados os resultados de simulações obtidos com o codificador modificado. É feita uma comparação entre o codificador proposto em [4], o codificador proposto e o padrão de compressão de vídeo H.264/AVC, cujo código fonte está disponível gratuitamente em [14]. A transformada *wavelet* implementada foi a transformada biortogonal 9/7 através do algoritmo de fatoração apresentado em [15]-[17]. Este algoritmo acelera o processamento do cálculo da transformada em até 50%. Devido a esta redução expressiva no esforço computacional, as transformadas *wavelet* implementadas por este método são denominadas *wavelets* de segunda geração [18]-[19]. Em todas as simulações foram utilizados os 60 primeiros quadros

monocromáticos (apenas a luminância foi processada) no formato CIF de 352 x 288 *pixels*, com 256 níveis de cinza (luminância com 8 *bits* de profundidade) das seguintes seqüências de vídeo: “Foreman”, “Susie”, “Table Tennis”, “Mobile and Calendar” e “Kiel”. Em cada seqüência há um tipo predominante de movimento de cena, desde efeitos de *zoom* na seqüência “Kiel”, como movimentos rápidos com movimentação da câmera na seqüência “Tennis”, movimentos de translação nas direções vertical e horizontal na seqüência “Mobile and Calendar” e pouca movimentação de cena com fundo estático, no caso das seqüências “Susie” e “Foreman”. Estas características permitem avaliar o desempenho do codificador quanto a sua robustez ao movimento de cena. O codificador H.264/AVC utilizado codifica as seqüências adotando um GOP com seis quadros formando a seqüência IBPBPB. As seguintes métricas de distorção foram adotadas para avaliar o desempenho do codificador:

- PSNR (*Peak Signal to Noise Ratio*). A PSNR dada pela equação 7, considerando que cada *pixel* das seqüências foram codificados com 8 bpp:

$$PSNR = 20 \log_{10} \left(\frac{255}{RMSE} \right). \quad (7)$$

- RMSE (*Root Mean Square Error*) é dado pela raiz quadrada do valor médio dos erros quadráticos entre os *pixels* do quadro original e do quadro decodificado. O RSME é dado pela equação 8:

$$RMSE = \sqrt{\frac{1}{N} \sum_k \sum_{m,n} (\varphi_1(m,n,k) - \varphi_2(m,n,k))^2}, \quad (8)$$

onde N é o número total de *pixels*, m e n são os índices das linhas e colunas, respectivamente, k é o número do quadro de cada seqüência, e φ_1 e φ_2 representam os quadros original e decodificado, respectivamente. Os valores obtidos de PSNR são valores médios para os 60 quadros processados de cada seqüência.

Nas simulações, a taxa de *bit* foi variada de 1 bpp até 0.0625 bpp, correspondendo a uma razão de compressão de 8 vezes e 128 vezes, respectivamente. A Fig. 8 apresenta a curva PSNR x taxa para a seqüência “Foreman”:

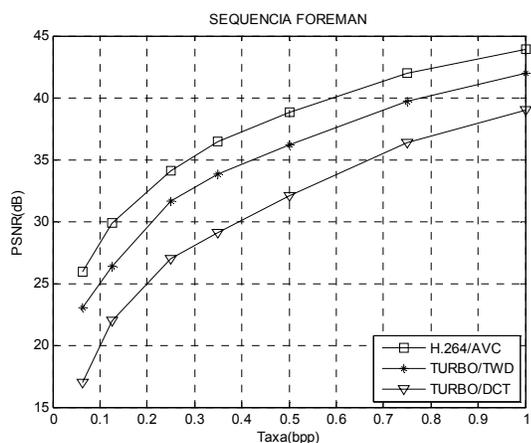


Fig. 8. Curva PSNR x taxa para a sequência "Foreman".

A sequência "Foreman" apresenta pouca movimentação de cena. Pela Fig. 8 observa-se um ganho no desempenho do codificador proposto de 2,9 a 6,0 dB em relação ao codificador de referência apresentado em [4]. O incremento no desempenho deve-se basicamente a melhor qualidade dos quadros pares (do inglês, *side information*) recuperados no decodificador pelo uso da transformada *wavelet* 9/7, já que a DCT manifesta efeito de bloco (do inglês, *blocking artifacts*) na decodificação a baixas taxas. Este efeito indesejável não está presente na transformada *wavelet* [20]. Para a sequência "Foreman", o codificador proposto encontra-se de 1,9 a 3,9 dB abaixo do H.264/AVC, mas com PSNR de 2,9 a 6,0 dB acima do codificador de referência.

A Fig. 9 apresenta os resultados obtidos para a sequência "Susie":

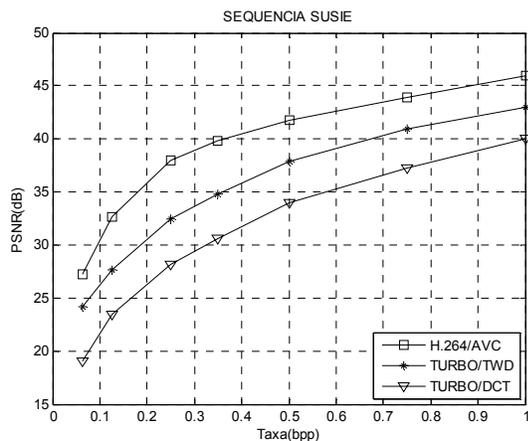


Fig. 9. Curva PSNR x taxa para a sequência "Susie".

A sequência "Susie" também apresenta pouca movimentação de cena. Nesta sequência, o plano de fundo da tela é estático e não há movimentação de câmera. Observa-se novamente o desempenho superior do codificador proposto em relação ao codificador que utiliza a transformada DCT, com um ganho de PSNR de 2,9 a 5,1 dB. O codificador proposto encontra-se de 2,8 a 5,5 dB abaixo do H.264/AVC.

A Fig. 10 apresenta os resultados para a sequência "Table Tennis":

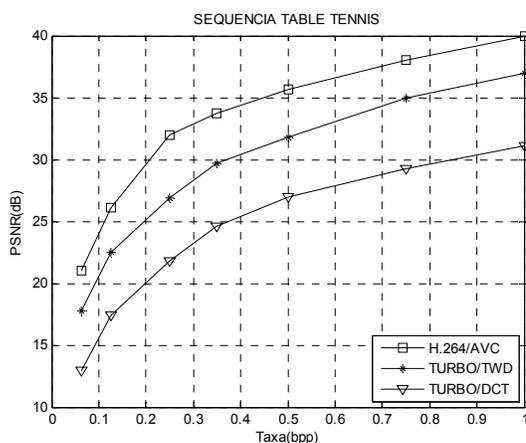


Fig. 10. Curva PSNR x taxa para a sequência "Table Tennis".

A sequência "Table Tennis" apresenta movimentação de cena mais complexa que as sequências "Foreman" e "Susie". Nessa sequência, há rotação e translação da câmera e movimentação rápida de objetos na cena, no caso, a bola de tênis de mesa. Para esta sequência, o codificador proposto encontra-se de 2,9 a 4,1 dB abaixo do H.264/AVC, mas apresenta PSNR de 4,7 a 5,9 dB acima do codificador de referência.

A Fig. 11 apresenta os resultados para a sequência "Kiel":

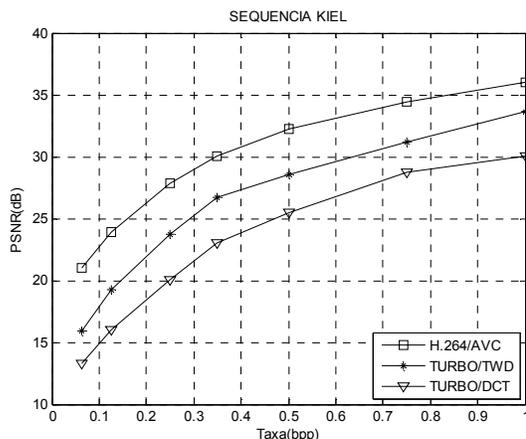


Fig. 11. Curva PSNR x taxa para a sequência "Kiel".

Esta sequência apresenta efeito de aproximação (do inglês, *zoom*) dos objetos de cena, uma característica que não está presente nas demais sequências analisadas. O codificador proposto possui PSNR de 2,7 a 3,5 dB superior ao codificador de referência e apresenta PSNR de 2,3 a 5,1 dB inferior em relação ao H.264/AVC.

A Fig. 12 apresenta os resultados para a sequência "Mobile and Calendar":

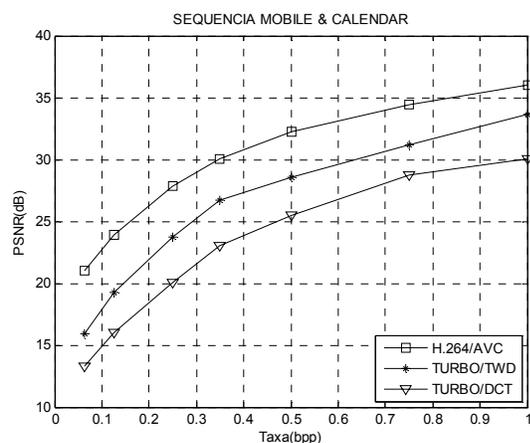


Fig. 12. Curva PSNR x taxa para a seqüência "Mobile and Calendar".

Esta seqüência possui a movimentação de cena mais complexa de todas as seqüências apresentadas, apresentando movimentos de rotação e translação de objetos e da câmera. O codificador proposto apresentou PSNR de 2,4 a 3,7 dB superior ao codificador de referência. A PSNR neste caso está de 2,4 a 5.1 dB inferior ao H.264/AVC.

IV. CONCLUSÕES

Neste trabalho, foi implementado um codificador de vídeo digital que utiliza códigos turbo e *wavelets* de segunda geração. O codificador proposto está baseado na teoria da codificação distribuída de fontes correlacionadas, um novo paradigma em compressão de vídeo digital. Este algoritmo permite a redução de até 100 vezes no tempo de processamento no codificador em relação ao H.264/AVC. Desta forma, o *codec* proposto torna-se uma alternativa promissora em dispositivos com limitações de processamento e memória. Os resultados de simulação apresentados comprovam o potencial do codificador em relação ao compressor estado da arte H.264/AVC.

REFERÊNCIAS

- [1] R. T. D. Pietro, "Algoritmo de casamento de padrões aplicado na estimação de movimento em compressão de vídeo," Tese de Doutorado, FEEC-UNICAMP, Janeiro 2003.
- [2] T. Sikora, "MPEG digital video coding standards," *IEEE Signal Processing Mag.*, pp. 82-100, Setembro 1997.
- [3] G. I. Sullivan and T. Wiegand, "Rate-distortion optimization for video coding compression," *IEEE Signal Processing Mag.*, pp. 74-90, Nov. 1999.
- [4] B. Girod, A. M. Aaron, S. Rane and D. Rebollo-Monedero, "Distributed video coding," Invited Paper in *Proc. IEEE*, Vol. 93, NO. 1, Jan. 2005.
- [5] J. Ascenso, C. Brites and F. Pereira, "Improving frame interpolation with spatial motion smoothing for pixel domain distributed source coding," *5th EURASIP Conference on Speech and Image Processing*, Slovak Republic, July 2005.
- [6] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Information Theory*, Vol. IT-19, pp. 471-480, July 1973.
- [7] A. D. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Information Theory*, Vol. IT-20, pp. 2-10, Jan. 1974.
- [8] A. D. Wyner, "On Source Coding with Side Information at the Decoder," *IEEE Trans. Information Theory*, Vol. IT-21, pp. 294-300, May 1975.

- [9] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information," *IEEE Trans. Information Theory*, Vol. IT-22, pp. 1-10, Jan. 1976.
- [10] J. E. Vila-Forcen, O. Kival and S. Voloshynovskiy, "Distributed single source with side information," *Proceedings of the SPIE*, Vol. 53, NO. 08, pp. 1060-1072, Jan. 2004.
- [11] A. Aaron and B. Girod, "Compression with side information using turbo codes," in *Proc. IEEE Data Compression Conf.*, pp. 252-261, April 2002.
- [12] Xiong, A. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Processing Magazine*, Vol. 21, pp. 80-94, Sept. 2004.
- [13] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, New York, Wiley, 1991.
- [14] Joint Video Team, "H.264 Reference Software," [ftp://ftp.itc.it/jvt-experts/reference software](ftp://ftp.itc.it/jvt-experts/reference%20software), Dec. 2002.
- [15] I. Daubechies, W. Sweldens, "Factoring Wavelet Transforms into Lifting Steps", Applied and Computational Mathematics, Princeton University, Princeton NJ, 1997.
- [16] G. Fernández, S. Periaswamy, W. Sweldens, "LIFTPACK: A Software Package for Wavelet Transforms using Lifting", Department of Computer Science, University of South Carolina, Columbia, 1996.
- [17] A. F. Ponchet, L. C. Martini, V. I. B. Sablón and Y. Iano, "Wavelets de segunda geração na compensação e estimação de movimentos do sinal de vídeo digital," *Revista Ciência e Tecnologia*, São Paulo, Vol. 1, pp. 31-39, Março 2004.
- [18] W. Sweldens, "Building Your Own Wavelets at Home," Department of Computer Science, Katholieke Universiteit Leuven, Belgium, 1996.
- [19] A. R. Calderbank, I. Daubechies, W. Sweldens, B. Yeo, "Wavelet Transforms that Map Integers to Integers", AT&T-Labs Research, Princeton University, 1996.
- [20] R. A. Devore, B. Jawerth, and B. Lucier, "Image Compression Through Wavelet Transform Coding", *IEEE Transactions on Information Theory*, vol. 38, pp. 719-746, 1992.
- [21] A. Skodras, C. Christopoulos and T. Ebrahimi, "The JPEG 2000 Still Image Compression Standard," *IEEE Signal Processing Magazine*, Vol. 18 NO. 5, pp. 36-58, Sept. 2001.
- [22] M. Rabbani and R. Joshi, "An overview of the JPEG2000 still image compression standard," *Signal Processing: Image Communication*, Vol. 17, NO. 1, pp. 3-48, Jan. 2002.
- [23] M. Vetterli, "Wavelets and Filter Banks: Theory and Design", *IEEE Trans. on Signal Processing*, Vol. 40, NO. 9, pp. 2207-2232, Sept. 1992.
- [24] S. G. Mallat, "A Theory of Multiresolution Signal Decomposition: The Wavelet Representation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, NO. 7, pp. 674-693, July 1989.
- [25] S. Givani, B. Penna, M. Grangetto, E. Magli and G. Olmo, "DSP performance comparison between lifting and filter banks for image coding," *Signal Analysis and Simulation Group at Politecnico di Torino*, Jan. 2002.
- [26] S. Barua, K. A. Kotteri, A.E. Bell and J. E. Carletta, "Optimal Quantized Lifting Coefficients for the 9/7 Wavelet," *Proc. of the Intern. Conf. on Acoustics, Speech and Signal Proc.*, ICASSP-2004, Montréal, Canada, Sept. 2004.
- [27] D. Liang, L. Cheng and Z. Zhang, "General construction of wavelet filters via a lifting scheme and its application in image coding," *Society of Photo-Optical Instrumentation Engineers*, Vol. 42, NO. 7, pp. 1949-1955, July 2003.

André da Fontoura Ponchet recebeu o título de Engenheiro Eletricista pela Universidade Federal do Ceará em 2002 e o título de Mestre em Engenharia Elétrica pela UNICAMP em 2006.

Atualmente é aluno de Doutorado da FEEC/UNICAMP e suas áreas de interesse incluem: processamento digital de sinais, compressão de imagem e vídeo digitais e sistemas de comunicação móvel.

Yuzo Iano recebeu os títulos de Engenheiro Eletrônico, Mestre e Doutor em Engenharia Elétrica através da UNICAMP. Atualmente é Professor Adjunto do DECOM/FEEC/ UNICAMP. É o responsável pelo Laboratório de Comunicações Visuais do DECOM. Seus interesses estão em processamento digital de sinais de áudio e vídeo. Trabalha com pesquisas em televisão e em especial com a HDTV desde 1986.