

# CHAVEAMENTO ADAPTATIVO ENTRE MÉTODOS DE DECISÃO PARA ESCOLHA DA PRIMEIRA DIREÇÃO DE PROCESSAMENTO DO ALGORITMO EI-DCT

Flávia Magalhães Freitas, Abraham Alcaim

**Resumo** - A utilização do algoritmo de extrapolação EI - Extension Interpolation - na codificação orientada por objeto com DCT baseada em blocos tem mostrado excelentes resultados a taxas baixas e médias. Em trabalhos recentes foram utilizados dois métodos para a escolha da primeira direção de processamento do algoritmo EI-DCT. Ambos são baseados nas variâncias (um na máxima e outro na mínima) dos comprimentos dos segmentos do objeto. Ao contrário do que a literatura apresenta, foi mostrado que a utilização isolada de qualquer um desses dois métodos não consegue superar o outro em todas as situações. Este artigo propõe um esquema de chaveamento adaptativo entre essas duas técnicas de decisão e uma terceira (baseada na capacidade de compactação de energia das transformadas), que supera o desempenho de codificação em relação aos obtidos com os métodos isolados. A estratégia de chaveamento proposta é baseada em um indicador morfológico. O artigo apresenta, ainda, resultados relativos ao comportamento do esquema proposto, quando diferentes esquemas de partição em blocos do objeto a ser codificado são utilizados.

**Palavras-chave:** Codificação orientada por objeto, DCT baseada em blocos, indicador morfológico.

**Abstract** - The use of the EI - Extension Interpolation - padding technique in object-oriented block-based DCT has shown excellent results at low and medium bit rates. In recent works, two methods for selecting the first processing direction of the EI-DCT algorithm have been proposed. Both of them are based on the variances (one on the maximum and the other on the minimum) of object segments lengths. However, our experimental results have shown that the isolated use of any of these two methods is not able to outperform the other one in all situations. This paper presents an adaptive switching scheme which selects either one of these two decision strategies or a third one, which is based on the energy compaction of the transforms. Its coding performance is better than that obtained from the use of any of the isolated methods. The proposed switching strategy is based on a morphological feature. The paper also presents performance results of the proposed scheme for different methods of object partitioning into blocks.

**Keywords:** Object-based coding, block-based DCT, morphological features.

Flávia Magalhães Freitas é filiada à Pontifícia Universidade Católica de Minas Gerais (E-mail: flaviamagfreitas@pucminas.br)  
Abraham Alcaim é filiado à Pontifícia Universidade Católica do Rio de Janeiro (E-mail: alcaim@cetuc.puc-rio.br)

## 1. INTRODUÇÃO

A codificação orientada por objeto tem recebido uma atenção considerável no campo de processamento de imagens. Algumas das aplicações importantes são as bibliotecas digitais, a reprodução e a manipulação de objetos em uma cena de TV ou na produção de um filme, sistemas de multimídia interativa e vídeo-telefonia móvel. Nessa área, um grande esforço tem sido dedicado à melhoria do desempenho de codificação das transformadas orientadas por objeto. Há basicamente duas abordagens para processar a imagem: a primeira usa a DCT adaptativa à forma (SA-DCT) [2]-[7] e a segunda emprega esquemas de DCT baseada em blocos, após processar os blocos de contorno usando-se algoritmos de extrapolação [6]-[13]. Neste artigo será proposta uma nova estratégia que visa à melhoria de desempenho da DCT baseada em blocos associada ao algoritmo de extrapolação EI (*Extension/Interpolation*), que aqui será denominada EI-DCT. Como foi visto em [1], a EI-DCT tem se mostrado uma das técnicas mais eficazes para taxas baixas e médias. O algoritmo de extrapolação EI aplica um conjunto de transformadas 1D em uma das direções, seguidas de inserção de zeros no domínio da transformada e um outro conjunto de transformadas 1D inversas [8],[10]. A seguir, o mesmo procedimento é realizado na outra direção.

A direção sobre a qual se aplica o primeiro conjunto de transformadas 1D no algoritmo de extrapolação EI influencia o desempenho final do codificador. Alguns esquemas para a determinação da primeira direção de processamento do algoritmo EI baseiam-se exclusivamente na forma do objeto segmentado [7],[10]. Sendo assim, não necessitam transmitir qualquer informação paralela. Essas estratégias produzem coeficientes que não dependem da taxa de bits, permitindo codificações rápidas e fáceis para diferentes aplicações. A direção preferencial de processamento em um bloco de contorno específico é determinada pela variância dos comprimentos dos segmentos do objeto nesse bloco em ambas as direções. Seguindo essa abordagem, há duas estratégias possíveis: (1) MILV (MInimum Lengths Variance) [10], que escolhe a direção associada à menor das variâncias e (2) MALV (MAXimum Lengths Variances) [7], que seleciona a direção associada à maior variância.

Em [7] foi apresentado um critério que seleciona a direção preferencial que produz o menor produto  $emq \times nbits$ , onde  $emq$  é o erro médio quadrático e  $nbits$  é o número de bits usados no processo de codificação. Nos resultados apresentados em [7], o critério apresentou ganhos inferiores a 1 dB em relação à estratégia MALV, o que, obviamente, não compensa

as desvantagens relacionadas à sua implementação: uma elevadíssima carga computacional e principalmente, a necessidade do cálculo de um novo conjunto de coeficientes a cada vez que ocorrer mudança na taxa de bits. A transmissão de um bit adicional por bloco de contorno, de forma a informar ao decodificador a direção adotada como preferencial, é responsável pela redução da eficiência do algoritmo.

A literatura apresenta discussões contraditórias sobre qual das estratégias - MILV ou MALV - é preferível para determinar a primeira direção de processamento no algoritmo de extrapolação EI. De fato, nós observamos que para alguns objetos MALV é preferível a MILV, mas para outros, MILV é a melhor opção. Em um trabalho recente [14], buscou-se avaliar essas conclusões ambíguas através de uma análise criteriosa dessas estratégias, com base em indicadores que relacionem características morfológicas e de textura da imagem. A proposta deste artigo é exatamente um esquema híbrido, o qual foi desenvolvido tomando como ponto de partida as conclusões apresentadas em [14]. O esquema proposto seleciona adaptativamente o mais eficiente em média, dentre três métodos de decisão, para a escolha da primeira direção de processamento - horizontal ou vertical - do algoritmo EI-DCT. Essa escolha é feita para cada grupo de blocos de contorno apresentando números semelhantes de pixels do objeto, e a decisão é tomada com base em um indicador morfológico proposto em [1] para análise de codificadores de imagem orientados por objeto. Será avaliada, também, a influência dos algoritmos de partição em blocos da imagem, investigados em [6] e [18], sobre o desempenho final do esquema híbrido.

A Seção 2 deste artigo apresenta breves descrições do algoritmo de extrapolação EI e dos métodos de partição de imagem em blocos discutidos neste artigo. A Seção 3 apresenta as condições experimentais usadas ao longo do trabalho e os resultados relativos à aplicação das estratégias MALV e MILV ao algoritmo EI-DCT, segundo o indicador morfológico TNPO (Textura média dos blocos, de acordo com o Número de Pixels do Objeto). Para os casos em que é difícil estabelecer uma correlação entre esse indicador e os esquemas baseados nas variâncias dos comprimentos, propõe-se uma estratégia alternativa, chamada MACI (MAXimum Cumulative Energy Sum), que mede a capacidade de compactação de energia das transformadas. Essa estratégia está descrita na Seção 4, onde também são apresentadas comparações entre os desempenhos de MILV, MALV e MACES. O esquema híbrido proposto é descrito na Seção 5. A Seção 6 contém resultados de simulação, com discussões. Na Seção 7, avalia-se a possibilidade da utilização do esquema híbrido associado aos algoritmos de partição da imagem que tenham sido apontados como os mais eficientes em [18]. Finalmente, a Seção 8 é dedicada às conclusões do artigo.

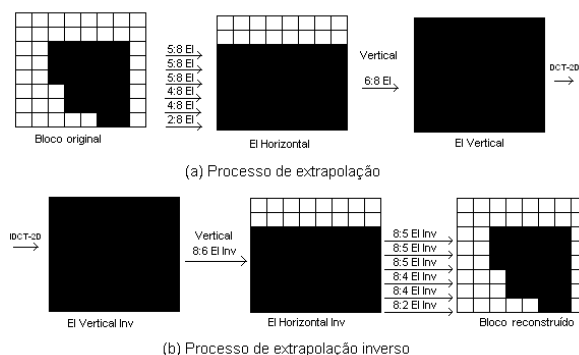
## 2. O ALGORITMO DE EXTRAPOLAÇÃO EI E ALGUNS MÉTODOS DE PARTIÇÃO DA IMAGEM EM BLOCOS

Na codificação de imagens orientada por objeto, uma opção atraente e alternativa à SA-DCT é o uso da

DCT baseada em blocos associada aos algoritmos de extrapolação [8]-[11], permitindo o uso direto dos algoritmos rápidos de DCT-2D. Nessa abordagem, os blocos de contorno são pré-processados, inicializando-se os pixels externos ao objeto antes da aplicação da DCT-2D.

Seguindo essa abordagem, em [8] foi apresentado um algoritmo de extrapolação para segmentos de imagem de forma arbitrária. Esse algoritmo foi denominado *Extension/Interpolation* (EI) e é constituído de dois procedimentos uni-dimensionais, o primeiro aplicado em uma direção e o segundo na outra.

Para ilustrar o algoritmo EI, consideremos que primeiro as operações sejam feitas na direção horizontal e depois, na direção vertical. Para cada uma das  $S$  linhas de objeto de tamanho  $M$  ( $M$  varia de acordo com a linha) em um bloco retangular de comprimento  $N$ , onde ( $1 \leq M < N$ ), calcula-se a DCT-1D sobre seus  $M$  pixels, obtendo-se  $M$  coeficientes da transformada. Em cada linha não nula, são inseridos  $(N - M)$  zeros no domínio da transformada e aplicada a DCT inversa sobre os  $N$  pontos, resultando  $N$  pixels interpolados no domínio espacial. Ao final do processamento na direção horizontal, resultarão  $S$  linhas de tamanho  $N$ . A seguir, o mesmo procedimento unidimensional é aplicado sobre as  $N$  colunas de tamanho  $S$ . A Figura 1 ilustra um exemplo do processo de extrapolação, na etapa de codificação, e do processo inverso de extrapolação, obrigatório do algoritmo EI, na etapa de decodificação.



**Figura 1.** Bloco de contorno extrapolado usando o algoritmo EI. (a) Processo de extrapolação (codificador) e (b) Processo de extrapolação inverso (decodificador)

Em qualquer das duas abordagens de codificação orientada por objeto, os blocos inteiramente contidos na região segmentada apresentam uma alta eficiência de codificação, mas isso não é necessariamente verdade nos blocos que contêm o contorno do objeto. Um procedimento viável visando à amenização desse problema consiste em encontrar uma técnica eficiente de partição em blocos da região que contém o objeto, de modo a reduzir o número de blocos a serem codificados [6]. Na EI-DCT, a redução do número de blocos a codificar significa um menor número de blocos a aplicar o algoritmo de extrapolação EI e um menor número de coeficientes a quantizar, o que reduz a complexidade e a taxa de

bits. Usando uma partição com essas características, pode-se conseguir ainda uma redução da carga computacional relativa à estimação e compensação de movimento baseadas em bloco, uma vez que o número de blocos que particionam o objeto é menor. Uma outra vantagem apontada para os esquemas de particionamento [6] é a necessidade da transmissão de um menor número de vetores de movimento para a predição de quadros na codificação inter-quadros (não contemplada neste artigo).

Nos métodos de compressão de imagem baseada em transformadas, a partição do quadro é feita em blocos de  $N \times N$  pixels e uma linha de blocos é denominada camada de blocos. Definindo-se como pixel de referência da camada o pixel superior esquerdo do primeiro bloco dessa camada, tem-se que no método tradicional de partição, o pixel de referência da primeira camada de blocos é sempre o pixel (1, 1), ou seja, o pixel superior esquerdo do quadro. Nos métodos de partição adaptativa à região segmentada, denominados métodos SARP (*Shape-Adaptive Region Partitioning*), a redução do número de blocos que particionam o objeto é obtida a partir da modificação das posições horizontais das camadas de blocos. Em [6] foram apresentados dois métodos SARP: o método ortogonal e o método flexível. No método ortogonal, os pixels de referência de todas as camadas de blocos estão localizados na mesma coluna e, portanto, há apenas uma referência horizontal a ser determinada e transmitida. O universo de busca do pixel de referência da primeira camada é o conjunto dos  $N^2$  pixels pertencentes ao primeiro bloco da primeira camada de blocos da partição convencional. O pixel escolhido como referência é aquele que, ao ser utilizado como referência para a primeira camada de blocos da partição, produz como resultado o menor número de blocos a serem codificados. Já no método flexível, as posições horizontais das camadas de blocos (colunas em que se localizam os respectivos pixels de referência) podem ser escolhidas de forma independente, sendo necessário que se determinem e transmitam referências horizontais distintas para cada uma dessas camadas. Evidentemente, apenas a referência vertical (linha que contém o pixel de referência) da primeira camada de blocos é calculada pelo algoritmo de partição, seja ele ortogonal ou flexível. As referências verticais das camadas subsequentes são obtidas somando-se  $N$  à referência vertical da camada anterior.

Os métodos ortogonal e flexível são implementados em duas versões distintas [6], que são diferentes quanto ao procedimento de busca dos pixels de referência. A primeira versão é chamada de busca completa, ou procedimento ótimo, no qual todos os  $N^2$  pixels candidatos à referência são testados. A segunda versão é chamada de busca parcial, ou procedimento simplificado, no qual se reduz o universo dos pixels a serem testados, visando à diminuição da carga computacional. No método ortogonal, o procedimento simplificado fixa uma referência horizontal candidata em 1 e calcula a referência vertical ótima para essa condição. Em seguida, essa referência vertical é tomada como definitiva e usada para a determinação da referência horizontal ótima final. O algoritmo flexível simplificado, por sua vez, mescla procedimentos do algoritmo ortogonal simplificado com parte do algoritmo flexível ótimo. Uma mesma referência horizontal can-

didata é tomada para todas as camadas de blocos e, variando-se a referência vertical candidata para a primeira camada (e por conseguinte, para as demais), calcula-se o número de blocos obtidos em cada camada. A referência vertical final escolhida para a primeira camada é aquela que produz o menor número total de blocos. Fixando-se essa referência vertical, calculam-se referências horizontais ótimas distintas para cada uma das camadas de blocos.

### 3. CONDIÇÕES EXPERIMENTAIS E APLICAÇÃO DAS ESTRATÉGIAS MILV E MALV AO ALGORITMO EI

Os trabalhos apresentados na literatura contemplando a aplicação das estratégias MALV e MILV para a determinação da direção preferencial do algoritmo de extrapolação EI são contraditórios ao apontar a supremacia absoluta de uma das estratégias sobre a outra [7],[10]. Nesses trabalhos, foram mostrados resultados para um repertório pequeno de imagens. Neste artigo o repertório de imagens foi ampliado, o que evidenciou que para algumas situações, MALV é superior a MILV, mas que para outras, o contrário é verificado.

A partir da constatação de que existe a incerteza sobre qual das duas estratégias é mais eficiente, buscou-se entender, através de um estudo experimental criterioso, em que condições uma das técnicas torna-se mais apropriada que a outra. Para uma maior precisão dos resultados, essa investigação deveria ser feita individualmente para cada bloco de contorno. Contudo, observamos que é possível estimar um comportamento médio para blocos apresentando números de pixels do objeto similares. Os resultados levaram ao indicador TNPO (Textura média dos blocos, de acordo com o Número de Pixels do Objeto), que apresenta uma boa correlação com o desempenho das estratégias MILV e MALV, quando as variâncias dos comprimentos dos segmentos do objeto nas direções vertical e horizontal são suficientemente distintas.

Na composição do indicador TNPO, os blocos de contorno foram divididos em 10 grupos, G1 a G10. Em G1 ficaram os blocos contendo de 1 a 7 pixels do objeto, em G2 blocos com 8 a 13 pixels, em G3 blocos com 14 a 19 pixels, em G4 blocos com 20 a 25 pixels, em G5 blocos com 26 a 32 pixels, em G6 blocos com 33 a 38 pixels, em G7 blocos com 39 a 44 pixels, em G8 blocos com 45 a 50 pixels, em G9 blocos com 51 a 56 pixels e em G10, blocos com 57 a 63 pixels do objeto. O desvio padrão das luminâncias dos pixels em cada bloco foi calculado e a média dos desvios padrões dos blocos pertencentes a um determinado grupo representa a textura média desse grupo, sendo armazenada no indicador TNPO.

Os experimentos foram realizados utilizando-se blocos de imagem de tamanho  $8 \times 8$ . Os blocos de contorno foram extrapolados usando o algoritmo EI associado às estratégias MALV e MILV e, finalmente, todos os blocos foram codificados usando a DCT-2D. Como medida de desempenho, foi utilizado o ganho em dB da razão pico/ruído (*RPR*) versus a taxa de bits por pixel do objeto (*bppo*). A *RPR* normalmente é calculada sobre a toda a região segmentada, mas, na fase de

calibração dos algoritmos, muitas vezes é calculada apenas sobre pixels do objeto que pertençam a determinado grupo de blocos de contorno (conforme indicador TNPO), sendo dada por

$$RPR = 10 \log_{10} \frac{VPP^2 * NPO}{emq} \quad (1)$$

onde  $VPP$  é o valor pico a pico da imagem de referência (no caso, 255),  $NPO$  é o número de pixels do objeto nos blocos de interesse e  $emq$  - erro médio quadrático - corresponde à soma dos quadrados dos erros de codificação dos pixels do objeto recuperado nesses blocos. A taxa usada para avaliação - bppo - corresponde à razão entre o número de bits alocados no processo de quantização e o número de pixels pertencentes aos blocos de interesse.

Reininger e Gibson [15] aplicaram testes de Kolmogorov-Smirnov (KS) aos coeficientes DCT de imagens e mostraram que a melhor aproximação para a distribuição dos coeficientes AC é a Laplaciana. Portanto, o coeficiente DC foi uniformemente quantizado com 8 bits e os coeficientes AC foram quantizados não uniformemente, com um número de bits dependente das respectivas variâncias [16]-[17], considerando uma distribuição de probabilidade laplaciana. A alocação de bits utilizada depende dos coeficientes finais da transformada, sendo expressa por:

$$B_i = B_0 + \frac{1}{2} \log_2 \frac{\sigma_i^2}{\left(\prod_{j=1}^{64} \sigma_j^2\right)^{\frac{1}{64}}} \quad i = 2, \dots, 64 \quad (2)$$

onde  $B_i$  é o número de bits alocados para o  $i$ -ésimo coeficiente AC,  $\sigma_i^2$  é a sua variância e  $B_0$  é um parâmetro que permite variar a taxa de bits média.

Para uma dada taxa  $B_0$  requerida para a codificação do objeto como um todo, foram computados os desempenhos de todos os blocos de contorno e também os resultados tomando-se apenas os blocos de um determinado grupo. Obviamente, para um dado  $B_0$ , os blocos pouco preenchidos apresentarão taxas de bppo mais altas, ao passo que os blocos muito preenchidos apresentarão baixas taxas de bppo.

O estudo experimental buscando decidir sobre a adequação de empregar MILV ou MALV foi realizado sobre um plano de objeto de vídeo (VOP - "Video Object Plane") de três seqüências com 11 quadros: WEATHER (144x176 pixels), CHILDREN (144x176 pixels) e FISH AND LOGO (240x352 pixels). Foram ainda usados um objeto dos quadros AKYIO (320x384 pixels), LENA (512x512 pixels) e PEPPER (512x512 pixels). A Figura 2 mostra o indicador morfológico TNPO para um objeto de AKYIO, CHILDREN (objeto 1), FISH AND LOGO (objeto 1), WEATHER, LENA e PEPPER (objeto 2). Esse indicador será relacionado à eficiência das estratégias MALV e MILV para a determinação da direção preferencial do algoritmo de extrapolação EI nos blocos de contorno (grupos G1 a G10). A partir do gráfico da Figura 2, constata-se que os objetos AKYIO e PEPPER apresentam baixas texturas para todos os grupos, ao passo que CHILDREN e FISH AND LOGO normalmente apresentam texturas mais elevadas. LENA e WEATHER apresentam, em geral, texturas de médias a altas, dependendo do grupo.

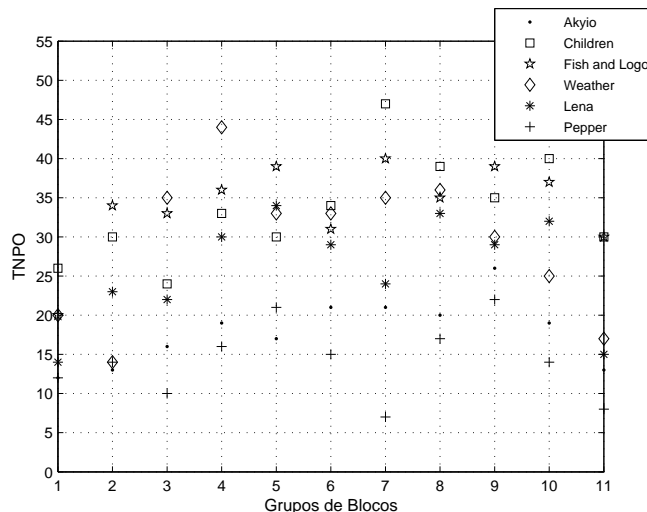


Figura 2. Textura de acordo com o Número de Pixels do Objeto - TNPO

Foi mostrado em [14] que na codificação de um objeto usando a EI-DCT, nem sempre a eficiência de MALV é melhor que a de MILV na determinação da direção de processamento preferencial, como havia sido afirmado em [7], e nem sempre MILV é superior a MALV, conforme havia sido concluído no trabalho [10]. Observou-se ainda que um mesmo objeto pode apresentar variações nos comportamentos médios dos desempenhos de MILV e MALV para grupos de blocos distintos, conforme a quantidade de pixels do objeto presentes, apontando para a possibilidade da escolha em separado da estratégia mais adequada a cada grupo. Foi verificado, também em [14], que para grupos com textura alta, a estratégia MILV normalmente é preferível à estratégia MALV. Para grupos de baixa textura, normalmente a estratégia MALV proporciona melhores ganhos.

É importante notar que a estratégia MALV seleciona como direção preferencial aquela na qual a maioria dos pixels do objeto está concentrada em segmentos mais longos. Esses segmentos têm a sua textura muito pouco suavizada pela aplicação do algoritmo de extrapolação EI, mas o algoritmo proporciona uma compactação de energia eficiente para blocos de textura baixa. Essa é a razão pela qual MALV é preferível nos blocos de baixa textura. Por outro lado, a estratégia MILV seleciona como direção preferencial aquela na qual os comprimentos dos segmentos do objeto estão mais uniformemente distribuídos. Esses segmentos normalmente são menores e portanto tendem a ser mais suavizados com a aplicação da técnica de extrapolação EI. Sendo assim, esse procedimento proporciona uma maior compactação de energia nos blocos de alta textura. Isso explica o desempenho superior de MILV para blocos de alta textura.

Contudo, para grupos de textura média e para grupos nos quais os blocos estejam praticamente preenchidos por pixels do objeto, implicando em diferenças pouco expressivas entre as variâncias dos comprimentos dos segmentos do objeto nas direções vertical e horizontal, a correlação entre o indicador TNPO e a eficiência das estratégias MILV e MALV é reduzida. Esses resultados sinalizam para a importância de se

pensar uma estratégia para a determinação da direção preferencial de processamento para os casos em que a correlação entre as eficiências das estratégias MILV e MALV e o indicador TNPO seja baixa. Obviamente, para que sua eficiência possa ser melhor relacionada ao indicador de textura do objeto, essa estratégia deverá ser baseada não apenas na forma do objeto, mas também nos coeficientes obtidos para o bloco ao se usar como preferencial ora a direção horizontal, ora a vertical. Mesmo que isso implique na transmissão de um bit extra para cada bloco, esse inconveniente poderá ser compensado pela melhoria da eficiência de codificação.

#### 4. A ESTRATÉGIA MACES

A análise apresentada em [14] buscava estabelecer, para cada grupo de blocos de contorno, faixas de textura que permitissem distinguir a adequação de empregar MILV ou MALV. Contudo, em algumas situações - como foi anteriormente observado - torna-se difícil determinar com precisão os limiares entre essas faixas. Nesses casos, não é possível apontar qualquer das duas estratégias como uma boa escolha. Portanto, torna-se necessário considerar uma estratégia diferente para determinar a primeira direção de processamento nesses casos. Uma possibilidade é usar não somente a forma do objeto, mas também a capacidade de compactação de energia dos coeficientes da DCT em um bloco de contorno extrapolado através do algoritmo EI usando cada uma das duas direções - horizontal ou vertical - como preferencial. Após uma investigação experimental cuidadosa de diversas medidas de compactação de energia nos casos em que MILV e MALV não podem ser usados eficientemente, propõe-se neste trabalho a estratégia *Maximum Cumulative Energy Sum* (MACES) como medida de desempenho nessas situações. Como será visto em seguida, em muitas situações essa estratégia é melhor que MILV e MALV em termos de correlação entre o indicador TNPO e a determinação da primeira direção de processamento do algoritmo de extrapolação EI. Isso ocorre à medida em que os blocos tornam-se mais cheios e apresentam, em média, valores de textura em uma faixa intermediária (entre 25 e 35, aproximadamente). A estratégia MACES, que será definida a seguir, também supera MILV nos casos de textura alta nos blocos pouco e medianamente preenchidos. Contudo, não é capaz de superar MALV e MILV em todas as situações, uma vez que necessita transmitir um bit de informação paralela adicional, por bloco de contorno.

A energia cumulativa em um bloco de contorno específico, que é uma medida usual na literatura, é dada por:

$$CE(k) = \sum_{m=0}^k [c(m)]^2 \quad (3)$$

onde  $\{c(0), c(1), \dots\}$  é o conjunto de todos os coeficientes DCT do bloco de imagem, em ordem decrescente de magnitude.

Para escolher a direção que proporciona a maior compactação de energia nos coeficientes finais da DCT, inicialmente foram comparados valores do vetor  $CE$  obtidos

ao usar-se ora a direção horizontal, ora a vertical como prioritária. Contudo, após um estudo experimental exaustivo, observou-se que para o objetivo visado neste trabalho, a soma das energias cumulativas dos coeficientes em ordem decrescente de magnitude é um indicador mais eficiente que apenas a energia cumulativa desses coeficientes. Sendo assim, para a determinação da direção preferencial de processamento, foi proposta a utilização da soma das componentes do vetor  $CE$  para  $k = 0, \dots, H$ . Essa investigação experimental permitiu escolher o valor de  $H$  correspondendo a 50% do número de coeficientes a quantizar em um bloco de contorno específico, o que equivale a 32 coeficientes, no caso da DCT baseada em blocos.

Sendo assim, a soma das energias cumulativas -  $CES$  - dos 32 coeficientes de maior magnitude é computada, para cada uma das duas direções, por:

$$CES = \sum_{k=0}^{31} CE(k) \quad (4)$$

A estratégia MACES seleciona como direção preferencial a que produzir o maior valor de  $CES$ . Esse esquema requer o cálculo de dois conjuntos de coeficientes da transformada e necessita da transmissão de um bit adicional por bloco de contorno para informar ao decodificador a direção selecionada.

Contudo, a vantagem dessa nova estratégia sobre a estratégia ótima no sentido  $emq \times nbits$  [7] é, principalmente, a utilização dos mesmos coeficientes no domínio das frequências espaciais, qualquer que seja a taxa de transmissão. Embora ambas as técnicas calculem dois conjuntos de coeficientes antes de decidir pela direção preferencial, a complexidade computacional dessa nova estratégia também é expressivamente menor, uma vez que elimina a necessidade de recuperação dos blocos codificados. Esse procedimento requer a aplicação das operações inversas de quantização, DCT-2D e extrapolação.

Os resultados de simulações mostradas nas Figuras 3 a 13, comparam, para alguns grupos de blocos, as eficiências das estratégias MILV, MALV e MACES. As Figuras 3 a 5 mostram o desempenho de codificação do grupo G2, no qual se nota que para objetos de textura baixa (aproximadamente até 30, como *AKYIO* e *LENA*), a estratégia MALV é a mais eficiente para a determinação da primeira direção de processamento do algoritmo EI. Contudo, os objetos de textura alta, como *FISH AND LOGO*, em geral são melhor codificados com a estratégia MACES. Nesses casos, essa estratégia, que mede a capacidade de compactação de energia nas duas direções, tende a ser ainda mais eficiente que MILV.

As Figuras 6 a 8 mostram os desempenhos de codificação dos blocos do grupo G4 para os objetos *AKYIO* (textura baixa), *CHILDREN* (textura média) e *WEATHER* (textura alta). A estratégia MALV permanece a mais eficiente para os casos de baixa textura, mas a estratégia MACES normalmente supera MALV e MILV em casos de textura alta. No caso de textura média, ocorre uma alternância entre os desempenhos de MILV e MALV, mas a estratégia MACES tende a aproximar ou superar a melhor delas.

Quando os blocos estão mais cheios, torna-se cada vez mais difícil apontar, segundo o indicador TNPO, qual a

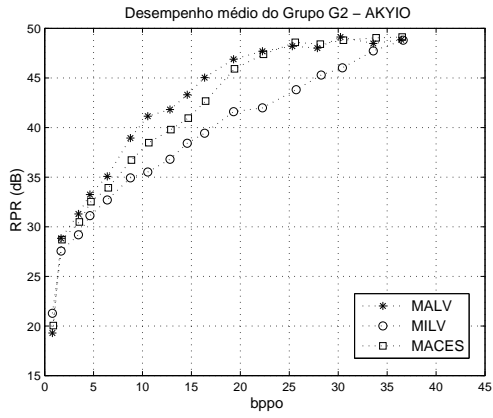


Figura 3. Resultados de codificação dos blocos do grupo G2 de AKYIO

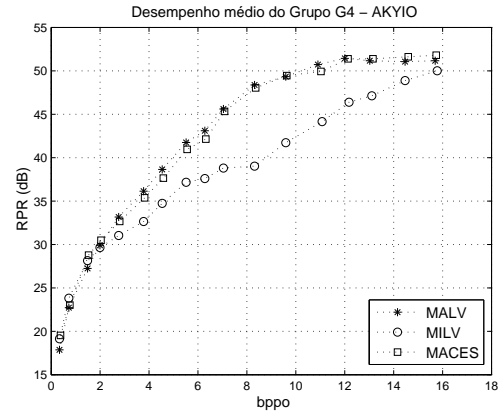


Figura 6. Resultados de codificação dos blocos do grupo G4 de AKYIO

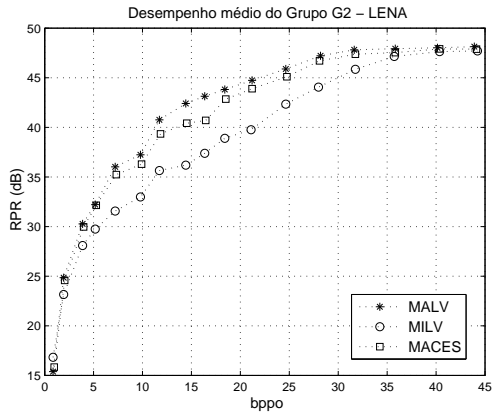


Figura 4. Resultados de codificação dos blocos do grupo G2 de LENA

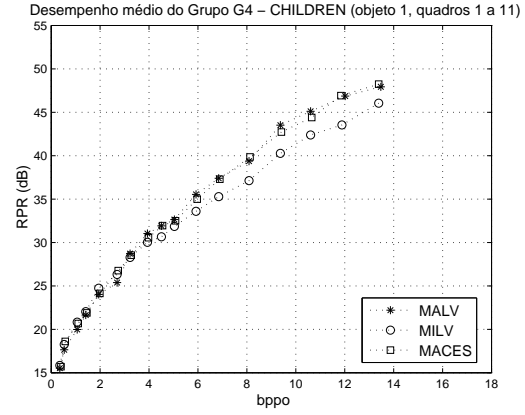


Figura 7. Resultados de codificação dos blocos do grupo G4 de CHILDREN (objeto 1, quadros 1 a 11)

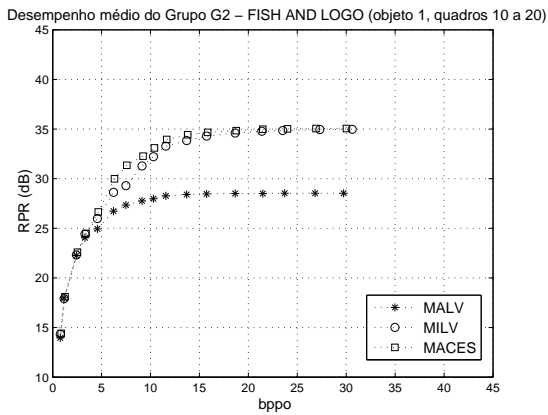


Figura 5. Resultados de codificação dos blocos do grupo G2 de FISH AND LOGO (objeto 1, quadros 10 a 20)

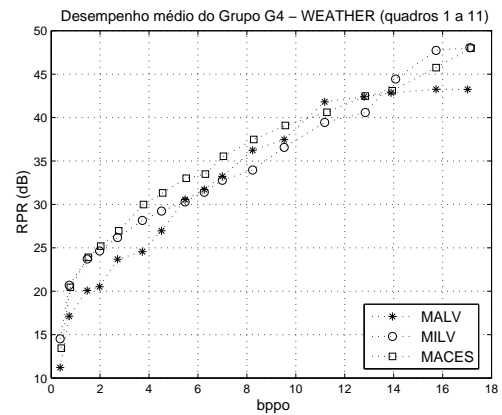


Figura 8. Resultados de codificação dos blocos do grupo G4 de WEATHER (quadros 1 a 11)

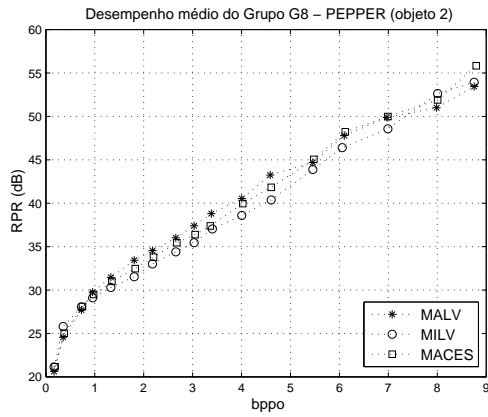


Figura 9. Resultados de codificação dos blocos do grupo G8 de PEPPER (objeto 2)

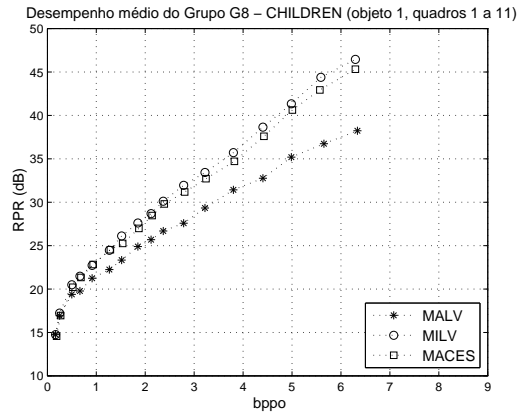


Figura 11. Resultados de codificação dos blocos do grupo G8 de CHILDREN (objeto 1, quadros 1 a 11)

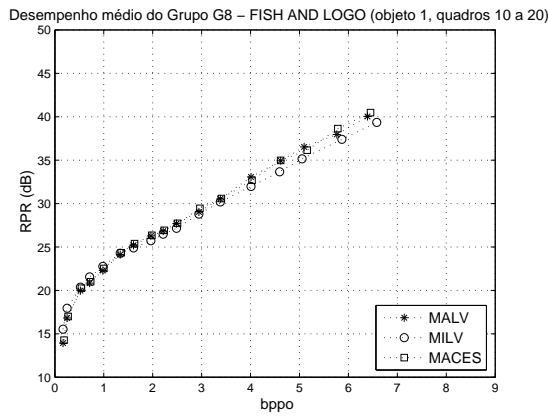


Figura 10. Resultados de codificação dos blocos do grupo G8 de FISH AND LOGO (objeto 1, quadros 10 a 20)

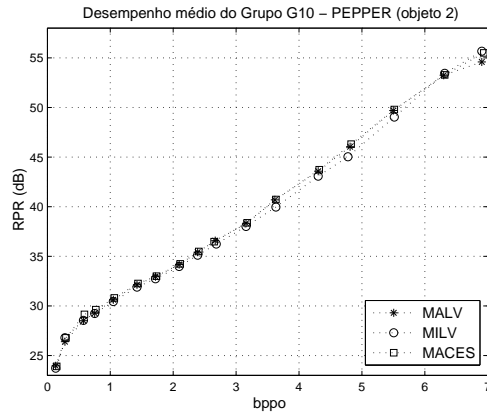


Figura 12. Resultados de codificação dos blocos do grupo G10 de PEPPER (objeto 2)

estratégia mais adequada entre MILV e MALV. Nessas condições, a estratégia MACES tende a estar mais relacionada ao indicador TNPO, a não ser em casos de textura extremamente baixa (para os quais normalmente MALV é melhor) ou muito alta (para os quais MILV é melhor). As Figuras 9 a 11 ilustram situações de textura muito baixa (PEPPER), média (FISH AND LOGO) e muito alta (CHILDREN) em G8, para as quais as estratégias MALV, MACES e MILV foram, respectivamente, as mais eficientes.

No grupo G10, os desempenhos de MILV, MALV e MACES são bastante semelhantes. Contudo, a estratégia MACES ainda tende a fornecer resultados um pouco superiores, principalmente porque as variâncias dos comprimentos em ambas as direções serão praticamente as mesmas e uma medida de capacidade de compactação de energia será mais adequada. Como exemplo, são mostradas nas Figuras 12 e 13, respectivamente, as curvas de desempenho das três estratégias no grupo G10 para os objetos PEPPER e FISH AND LOGO, cujas texturas nesse grupo podem ser classificadas como baixa e alta, respectivamente.

Uma síntese das observações feitas nesta seção é apresentada a seguir:

- MALV é mais indicada em grupos de baixa textura e

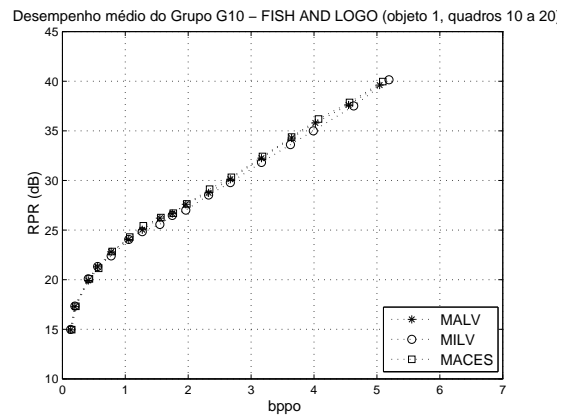


Figura 13. Resultados de codificação dos blocos do grupo G10 de FISH AND LOGO (objeto 1, quadros 10 a 20)

MILV, em grupos de alta textura. Contudo, MACES supera MILV em situações de texturas altas nos blocos pouco ou medianamente preenchidos (G1 a G6);

- MACES é mais adequada em todas as situações de texturas médias (G1 a G10), uma vez que os limiares de textura para o chaveamento entre MALV e MILV não são absolutamente precisos (um grupo com textura na faixa intermediária pode conter vários blocos com baixa textura e também vários blocos com alta textura, para os quais MALV e MILV, respectivamente, seriam mais indicados);
- MACES é a mais adequada nos blocos quase cheios (G10), no qual praticamente não existe diferença entre as variâncias dos comprimentos nas duas direções.
- MACES não supera MILV e MALV em todas as situações principalmente porque o ganho de codificação obtido com essa estratégia nem sempre compensa o aumento da informação paralela em um bit por bloco de contorno.

## 5. O ESQUEMA HÍBRIDO PROPOSTO PARA A EI-DCT

Foi observado que ao se aplicar somente MALV ou somente MILV a todos os blocos de contorno (conforme os trabalhos já publicados na literatura), essa escolha pode levar a resultados bons ou ruins, dependendo das características das imagens e dos blocos de contorno. Concluímos ainda que escolhendo a estratégia mais adequada (MALV ou MILV) para cada bloco de contorno de uma determinada imagem, os resultados podem ser comparáveis ou melhores que aqueles obtidos quando somente a melhor das estratégias (quando usada isoladamente para aquela imagem em particular) é empregada. Idealmente, a escolha da estratégia mais adequada - MILV ou MALV - deveria ser feita individualmente para cada bloco de contorno. Contudo, isso demandaria a transmissão de 1 bit adicional para cada bloco (o que seria igual à informação paralela exigida pelo esquema que seleciona a direção que apresenta o menor produto  $emq \times nbits$ , acarretando algumas das desvantagens óbvias desse tipo de procedimento). Por esse motivo, optamos por agrupar os blocos de contorno segundo o número de pixels do objeto contidos nesses blocos, e analisarmos a eficiência de uma ou outra técnica levando em conta a média das texturas dos blocos em cada grupo, através do indicador TNPO. Isso foi feito após ser observado que a eficiência de MALV e MILV pode, na maioria dos casos, ser relacionada ao número de pixels e à textura dos blocos. Se o indicador TNPO contemplasse um maior número de grupos, a precisão da escolha das estratégias seria mais refinada (o que aumentaria o ganho de codificação), mas a informação paralela seria maior (o que aumentaria a taxa de bits).

Vale ressaltar que no esquema híbrido proposto a estratégia MACES é usada apenas quando a correlação entre o indicador TNPO e o desempenho das estratégias MALV e MILV é reduzida ou então quando o ganho de desempenho de

MACES em relação à estratégia MILV compensa o aumento de um bit de informação por bloco.

A idéia básica do novo método para a escolha da direção preferencial na EI-DCT consiste em estabelecer, para cada grupo de blocos de contorno, limiares de textura que permitam distinguir a adequação de empregar cada um dos métodos de decisão: MILV, MALV ou MACES. Baseando-se no indicador TNPO, podem ser geradas regras para selecionar a estratégia mais adequada para cada grupo  $g \in \{G1, \dots, G10\}$ .

A escolha dos limiares do esquema de chaveamento foi uma abordagem puramente experimental, realizada com todo o rigor científico. Para a escolha da estratégia mais adequada a um grupo particular de blocos de contorno, foram computadas as medidas de desempenho  $RPR \times bppo$  obtidas com cada uma das técnicas nos blocos pertencentes a esse grupo, separadamente para cada um dos objetos de um repertório de imagens. As imagens utilizadas para uma primeira calibração do algoritmo foram: 11 quadros da seqüência CHILDREN (objeto 1), 11 quadros da seqüência FISH AND LOGO (objeto 1), 11 quadros da seqüência WEATHER e 1 quadro da seqüência AKYIO. Com essas imagens, foi possível estabelecer faixas de valores de textura que continham o valor mais adequado de chaveamento para cada grupo de blocos, sem, contudo, poder definir esses valores com muita precisão.

Em seguida, a calibração do esquema híbrido foi refinada pela inclusão do segundo objeto da seqüência CHILDREN e dos outros dois objetos da seqüência FISH AND LOGO, o que aumentou significativamente o número de blocos em cada um dos grupos de blocos de contorno - G1 a G10, permitindo a escolha, dentro das faixas inicialmente estabelecidas, dos valores mais adequados de chaveamento. Essa definição levou em conta basicamente dois aspectos: (1) os valores de chaveamento deveriam ser tais que levassem à escolha da melhor estratégia, em média, para todos os blocos de um determinado grupo e (2) o algoritmo deveria ser o mais simples possível, com a utilização de uma mesma regra para o maior número possível de grupos. Após essa calibração mais refinada, foram utilizadas outras imagens para a corroboração dos resultados: LENA, FIGHTER e 2 objetos de PEPPER.

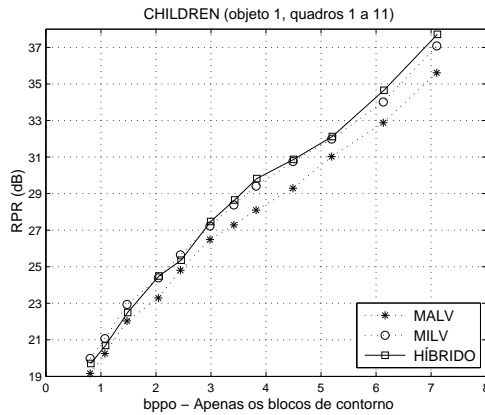
Uma vez que uma mesma estratégia é escolhida para todos os blocos de um grupo específico, não é necessário transmitir um bit adicional por bloco sempre que for selecionada a estratégia MILV ou a estratégia MALV. E ainda, não é necessário transmitir a textura média de cada grupo. Da Tabela 1, é claro que apenas 1 bit por grupo precisa ser transmitido para os grupos G1 a G6 e 2 bits por grupo para G7 a G9. Portanto, para implementar o esquema híbrido, somente 12 bits extras são usados, em adição ao bit adicional requerido por cada bloco utilizando a estratégia MACES.

O esquema híbrido proposto, além de ser um algoritmo de codificação orientado por objeto, também é orientado por grupos de blocos. Para que os resultados obtidos possam ser generalizados, é desejável que o número de blocos codificados seja expressivo para todos os grupos de blocos. A Tabela 2 apresenta o número de blocos em cada um dos grupos (G1 a G11) para os objetos codificados neste artigo.



**Tabela 1.** O Algoritmo Híbrido

$g = G1$ a $G5$	$g = G6$
MACES, se $TNPO(g) \geq 31$ MALV, se $TNPO(g) < 31$	MACES, se $TNPO(g) \geq 21$ MALV, se $TNPO(g) < 21$
$g = G7$ a $G9$ MILV, se $TNPO(g) \geq 36$ MACES, se $36 > TNPO(g) \geq 24$ MALV, se $TNPO(g) < 24$	$g = G10$ MACES



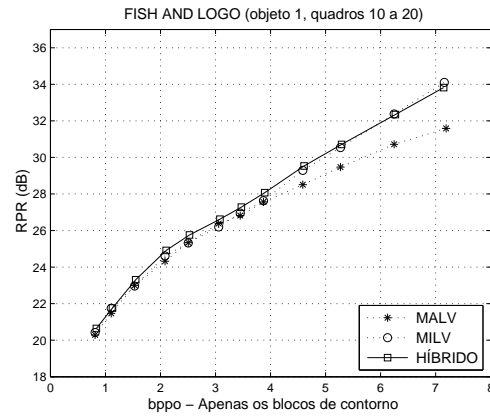
**Figura 14.** Desempenho das diversas estratégias EI-DCT para os 11 quadros do objeto 1 de *CHILDREN*

## 6. RESULTADOS DE SIMULAÇÃO

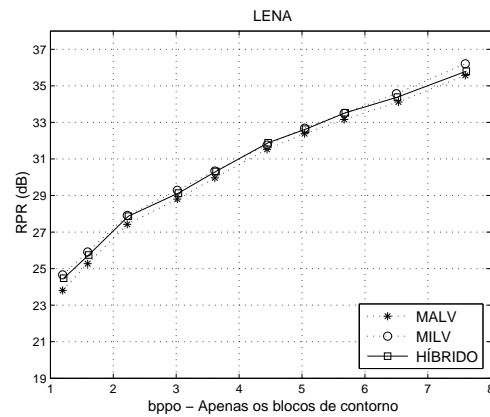
Os resultados que serão mostrados a seguir referem-se à codificação dos objetos e às condições descritas na Seção 3. Os blocos de contorno foram codificados usando a DCT baseada em blocos após terem sido processados pelo algoritmo de extrapolação EI. A primeira direção de processamento foi selecionada por três diferentes estratégias: MILV, MALV e o esquema HÍBRIDO. A medida de desempenho é dada em termos das curvas de  $RPR \times bpp$  e os valores de  $RPR$  foram medidos apenas sobre os blocos de contorno.

As Figuras 14 a 19 apresentam as curvas de  $RPR \times bpp$  para cada objeto. Para o cômputo da taxa  $bpp$ , considerou-se a informação paralela necessária à implementação do esquema híbrido. Pode ser notado que MILV é preferível a MALV em *CHILDREN*, *FISH AND LOGO* e *LENA*. Por outro lado, MALV é preferível a MILV em *WEATHER*, *AKYIO* e *PEPPER*. Isso mostra que não é uma boa opção usar somente MILV, como sugerido em [10], ou MALV, como proposto em [7]. Nossos resultados também apontaram que, independente de qual das estratégias é mais adequada para um determinado objeto, o algoritmo híbrido foi comparável, e em alguns casos superior, ao melhor deles.

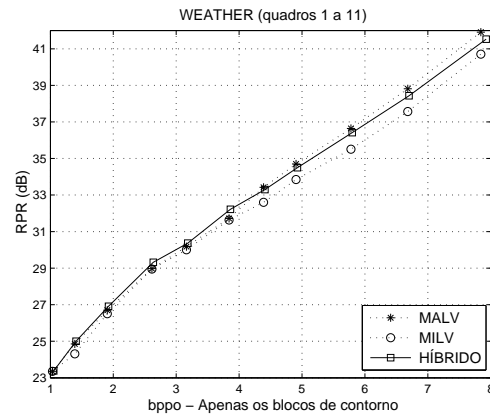
O algoritmo proposto seleciona de forma independente a



**Figura 15.** Desempenho das diversas estratégias EI-DCT para os 11 quadros do objeto 1 de *FISH AND LOGO*



**Figura 16.** Desempenho das diversas estratégias EI-DCT para o objeto de *LENA*



**Figura 17.** Desempenho das diversas estratégias EI-DCT para os 11 quadros do objeto de *WEATHER*

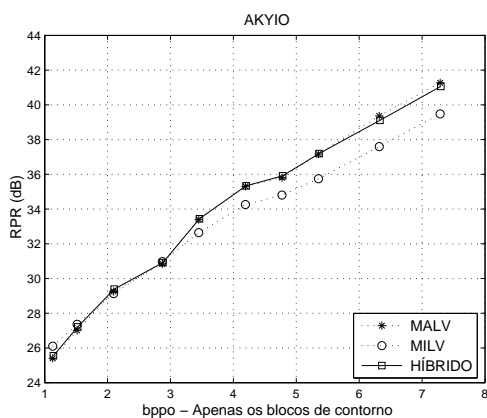


Figura 18. Desempenho das diversas estratégias EI-DCT para o objeto de AKYIO

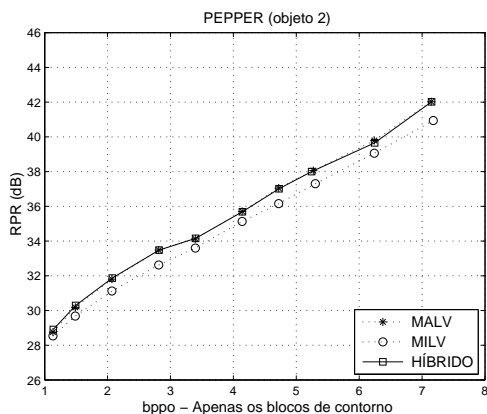


Figura 19. Desempenho das diversas estratégias EI-DCT para o objeto 2 de PEPPER

Tabela 2. Número de blocos por grupo em cada um dos objetos

OBJETO	G1	G2	G3	G4	G5	G6
AKYIO	16	13	4	7	9	3
CHILDREN						
(obj. 1)	55	22	32	46	27	28
(obj. 2)	70	52	51	26	25	38
FISH						
(obj. 1)	89	48	44	51	45	36
(obj. 2)	151	82	48	32	38	32
(obj. 3)	84	44	38	38	41	38
WEATHER	57	54	54	15	35	20
LENA	24	24	26	13	17	9
FIGHTER	21	6	6	18	21	5
PEPPER						
(obj. 2)	21	14	15	9	11	5
(obj. 3)	16	11	15	7	25	10
TOTAL	604	370	333	262	294	224
OBJETO	G7	G8	G9	G10	G11	
AKYIO	5	6	7	17	550	
CHILDREN						
(obj. 1)	37	31	26	27	157	
(obj.2)	37	51	40	57	213	
FISH						
(obj. 1)	37	36	46	79	290	
(obj. 2)	32	49	36	73	232	
(obj. 3)	29	47	36	64	224	
WEATHER	33	37	53	48	1003	
LENA	8	13	14	30	1906	
FIGHTER	14	15	7	11	451	
PEPPER						
(obj. 2)	20	10	9	19	687	
(obj.3)	16	5	7	23	794	
TOTAL	268	300	281	448	6507	

melhor estratégia para cada grupo. Isso é possível porque essa estratégia adaptativa é capaz de explorar, de modo eficiente, as características de forma, textura e compactação de energia nos blocos de contorno.

## 7. RESULTADOS COM PARTICIONAMENTOS ORTOGONAL E FLEXÍVEL

Nesta seção, aplicaremos os algoritmos de partição da imagem em blocos [6] que foram apontados em [18] como as melhores soluções de compromisso em termos de capacidade de redução do número de blocos, complexidade de implementação e tempo de processamento. Esses algoritmos foram o Ortogonal Simplificado e o Flexível Simplificado, descritos na Seção 2. Ambos têm por objetivos principais a redução do número total de blocos a codificar (*NTB*) e do número de blocos de contorno (*NBC*), levando a blocos de contorno mais cheios, que tendem a apresentar maiores desempenhos de codificação. Conjuntamente, esses critérios aumentam a Área Efetiva (*AE*) dos objetos, um indicador

morfológico também definido em [1] e que equivale à porcentagem da área que efetivamente pertence à região segmentada, dentro daquela ocupada pelos blocos que contêm pixels do objeto. A Área Efetiva é calculada por:

$$AE = \frac{NPO}{64 \times NTB} \times 100\% \quad (5)$$

em que  $NPO$  é o número de pixels do objeto. O algoritmo Flexível Simplificado reduz as quantidades  $NTB$  e  $NBC$  cerca de duas vezes mais que o algoritmo Ortogonal Simplificado, o que produz um aumento considerável da Área Efetiva. Como o número de pixels a codificar na DCT baseada em blocos é igual a  $64 \times NTB$ , isso representa a necessidade de menos bits para uma dada especificação de qualidade.

A Tabela 3 apresenta os valores da Área Efetiva para os objetos codificados sob os particionamentos Convencional ( $C$ ), Ortogonal Simplificado ( $OS$ ) e Flexível Simplificado ( $FS$ ). Também são mostrados os valores da Área Efetiva ao se excluir, da região segmentada, os blocos inteiros. Chamaremos essas quantidades de  $AE_{BC}$ , que têm a função de indicar quão cheios estão os blocos de contorno de um objeto específico e podem ser obtidas através da expressão (5), substituindo-se a quantidade  $NTB$  por  $NBC$ .

**Tabela 3.** Áreas Efetivas dos objetos sob diferentes esquemas de particionamento

OBJETOS	AE (todos os blocos)			AE <sub>BC</sub> (apenas blocos de contorno)		
	C	OS	FS	C	OS	FS
CHILDREN	63,8	65,6	72,0	46,7	48,9	57,0
FISH	66,8	69,4	72,8	48,0	49,0	52,6
WEATHER	85,1	86,2	90,3	48,4	39,5	50,8
AKYIO	93,0	93,3	95,0	48,5	45,8	54,0
LENA	95,5	95,7	96,7	47,1	47,6	54,0
PEPPER	91,6	92,9	94,1	48,3	49,8	54,2

A partir da Tabela 3, observa-se que o algoritmo Flexível Simplificado sempre aumenta os valores de  $AE$  e  $AE_{BC}$ , isto é, além de aumentar a Área Efetiva do objeto como um todo, reduzindo o número total de blocos, também produz blocos de contorno mais cheios que os blocos de contorno da partição convencional. Já com o algoritmo Ortogonal Simplificado, o aumento da Área Efetiva é bem inferior, além de produzir, em alguns casos como *WEATHER* e *AKYIO*, blocos de contorno em média mais vazios que os da partição convencional. Nota-se também que os objetos *AKYIO* e *LENA* foram os menos beneficiados com a aplicação do particionamento Flexível Simplificado, uma vez que já apresentavam, com a partição convencional, um alto valor de Área Efetiva.

Cabe ressaltar que a mudança no esquema de particionamento não altera apenas a Área Efetiva. O indicador  $TNPO$  de cada objeto também muda sensivelmente, produzindo uma distribuição de blocos pelos diversos grupos completamente distinta. Isso algumas vezes provoca, inclusive, a alteração da estratégia escolhida para a determinação da direção preferencial em um determinado grupo. Contudo, os resultados obtidos mostraram a eficiência do algoritmo proposto para essa

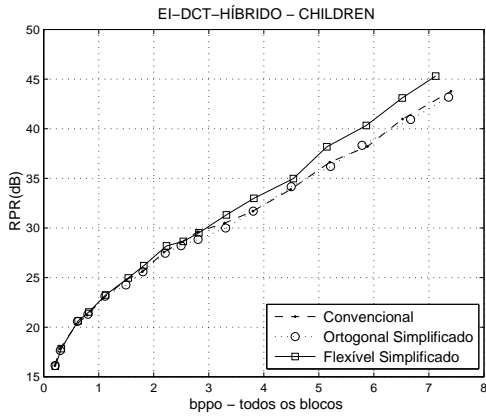
nova distribuição de blocos e indicadores  $TNPO$ , comprovando a boa escolha dos limiares de chaveamento. As Tabelas 4 e 5 apresentam o número de blocos em cada um dos grupos (G1 a G11) para os objetos codificados, ao utilizarem-se os particionamentos Ortogonal Simplificado e Flexível Simplificado. A Tabela 6, por sua vez, mostra o número de blocos total, somando-se os blocos obtidos com os particionamentos convencional, ortogonal e flexível simplificados. Observa-se que o número de blocos codificados foi expressivo para todos os grupos, o que enfatiza a validade do esquema proposto.

**Tabela 4.** Número de blocos por grupo em cada um dos objetos - Particionamento Ortogonal Simplificado

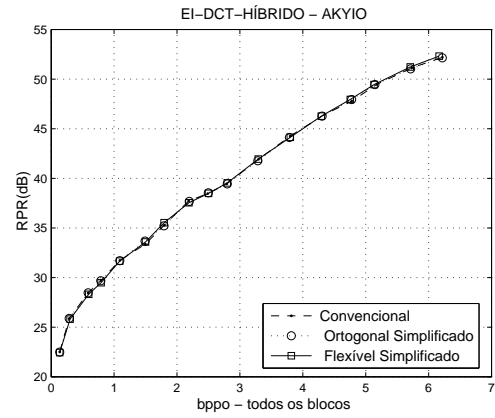
OBJETO	G1	G2	G3	G4	G5	G6
AKYIO	16	8	5	9	5	8
CHILDREN						
(obj. 1)	51	32	28	25	27	21
(obj. 2)	41	40	43	37	41	17
FISH						
(obj. 1)	69	51	43	34	40	35
(obj. 2)	123	73	43	38	36	41
(obj. 3)	67	26	52	33	41	33
WEATHER	43	83	32	31	29	16
LENA	21	17	34	17	9	12
FIGHTER	11	6	9	16	25	8
PEPPER						
(obj. 2)	11	14	11	16	13	4
(obj. 3)	12	14	12	6	12	7
TOTAL	465	364	312	262	278	202
OBJETO	G7	G8	G9	G10	G11	
AKYIO	3	8	8	9	556	
CHILDREN						
(obj. 1)	34	33	30	39	155	
(obj.2)	23	54	37	53	231	
FISH						
(obj. 1)	50	36	41	63	309	
(obj. 2)	33	48	34	63	242	
(obj. 3)	33	29	42	58	241	
WEATHER	11	23	26	24	1074	
LENA	10	9	15	28	1908	
FIGHTER	9	8	8	11	456	
PEPPER						
(obj. 2)	8	8	18	12	694	
(obj.3)	7	12	8	18	807	
TOTAL	221	268	267	378	6673	

A partir dos resultados mostrados nas Figuras 20 a 25, que contêm os desempenhos do esquema híbrido utilizando-se os particionamentos Convencional, Ortogonal Simplificado e Flexível Simplificado para os objetos codificados, percebe-se que a eficiência do esquema híbrido pode ser aumentada (em alguns casos, sensivelmente) ao se utilizar o particionamento Flexível Simplificado.

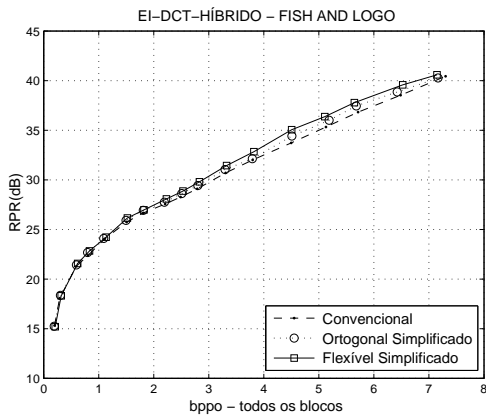
Os gráficos das Figuras 26 a 28 mostram os ganhos médios de  $RPR$  apresentados pelas estratégias MILV e HÍBRIDA sobre a estratégia MALV nos particionamentos



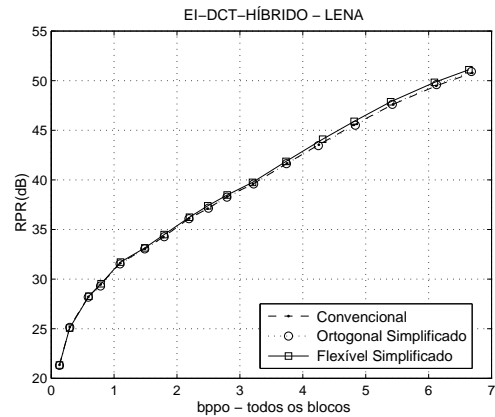
**Figura 20.** Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *CHILDREN* (objeto 1)



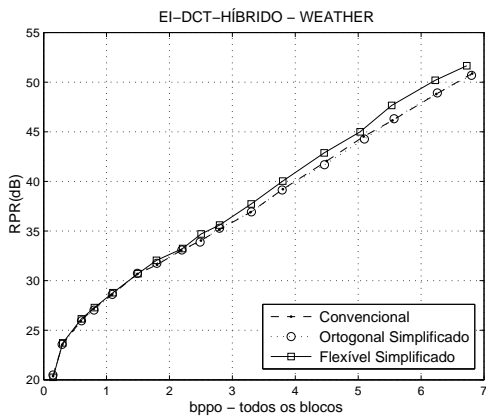
**Figura 23.** Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *AKYIO*



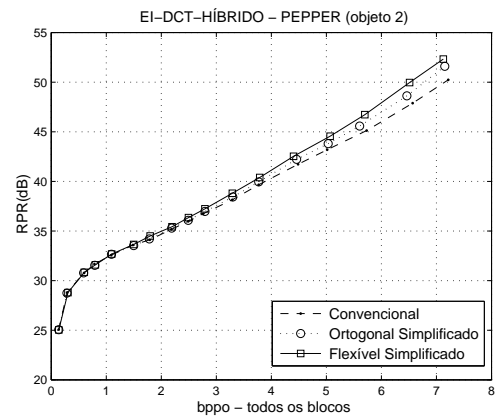
**Figura 21.** Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *FISH AND LOGO* (objeto 1)



**Figura 24.** Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *LENA*



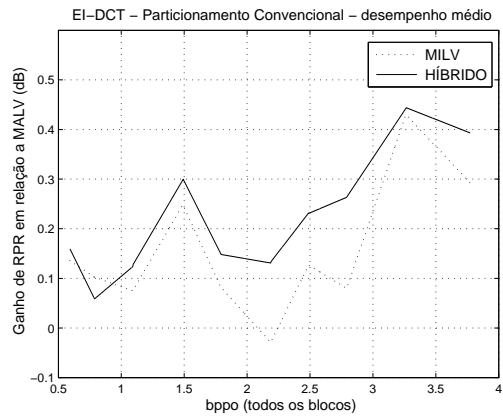
**Figura 22.** Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *WEATHER*



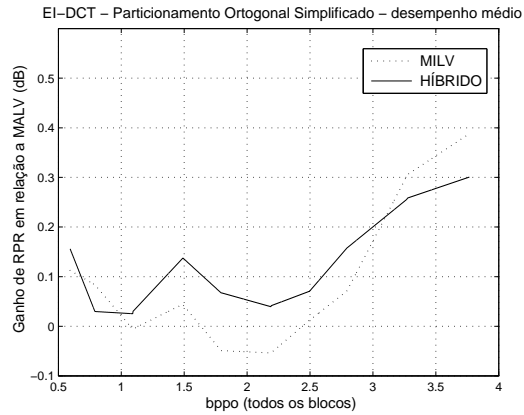
**Figura 25.** Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *PEPPER* (objeto 2)

**Tabela 5.** Número de blocos por grupo em cada um dos objetos - Particionamento Flexível Simplificado

OBJETO	G1	G2	G3	G4	G5	G6
AKYIO	4	7	8	7	7	3
CHILDREN						
(obj. 1)	17	19	29	33	16	20
(obj. 2)	12	18	26	24	40	33
FISH						
(obj. 1)	36	37	51	31	49	39
(obj. 2)	71	68	43	42	37	47
(obj. 3)	22	35	55	39	32	39
WEATHER	7	38	33	23	41	26
LENA	8	10	22	14	16	17
FIGHTER	4	6	10	16	23	7
PEPPER						
(obj. 2)	4	9	7	20	13	6
(obj. 3)	3	10	9	13	6	7
TOTAL	465	364	312	262	278	202
OBJETO	G7	G8	G9	G10	G11	
AKYIO	4	11	4	12	556	
CHILDREN						
(obj. 1)	34	32	35	48	150	
(obj.2)	45	38	49	45	232	
FISH						
(obj. 1)	35	41	47	57	313	
(obj. 2)	45	40	36	44	255	
(obj. 3)	41	36	40	31	256	
WEATHER	21	22	20	32	1066	
LENA	8	10	15	26	1911	
FIGHTER	17	9	7	6	457	
PEPPER						
(obj. 2)	6	10	16	11	696	
(obj.3)	10	12	10	12	812	
TOTAL	266	261	279	324	6704	



**Figura 26.** Ganhos de desempenho médios sobre a estratégia MALV no Particionamento Convencional

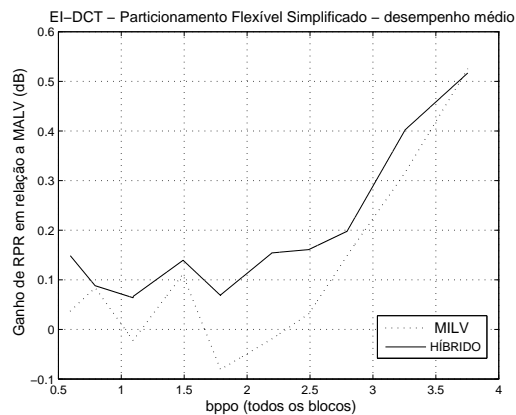


**Figura 27.** Ganhos de desempenho médios sobre a estratégia MALV no Particionamento Ortogonal Simplificado

Convencional, Ortogonal Simplificado e Flexível Simplificado, considerando todos os objetos citados na Seção 5. Para a composição dessas curvas, tomou-se o valor médio de codificação de todos os blocos dos seis objetos. Nota-se que em geral a eficiência do algoritmo HÍBRIDO foi superior à de MILV e MALV nos três particionamentos. A única exceção ocorreu para o particionamento Ortogonal Simplificado a taxas acima de 3,1 bppo, em que a estratégia MILV mostrou-se superior à estratégia HÍBRIDA. Contudo, vale ressaltar que a DCT baseada em blocos é mais aplicada a taxas baixas. A taxas altas, a SA-DCT é, reconhecidamente, mais eficiente, podendo ser usada preferencialmente à estratégia HÍBRIDA apresentada.

## 8. CONCLUSÕES

Este artigo apresentou um trabalho experimental criterioso, baseado em um estudo detalhado da literatura, no qual se buscou, além de mostrar as contradições em trabalhos já publi-



**Figura 28.** Ganhos de desempenho médios sobre a estratégia MALV no Particionamento Flexível Simplificado

**Tabela 6.** Número total de blocos por grupo nos Particionamentos Convencional, Ortogonal Simplificado e Flexível Simplificado

GRUPOS	G1	G2	G3	G4	G5	G6
TOTAL GERAL	1257	991	938	786	852	670
GRUPOS	G7	G8	G9	G10	G11	
TOTAL GERAL	755	829	827	1150	19884	

cados, propor uma solução efetiva para o problema e também investigar a sua aplicação em situações de particionamento adaptativo de blocos.

Inicialmente, foi apresentado o comportamento de duas estratégias reportadas na literatura para a escolha da direção preferencial de processamento do algoritmo EI, na codificação de imagens baseada em objeto: MALV e MILV. Esses esquemas são baseados nas variâncias dos comprimentos dos segmentos do objeto nas direções horizontal e vertical. MALV seleciona a direção com a máxima variância e MILV, a direção com a mínima variância. Em [10] o algoritmo MILV foi escolhido como o mais adequado para decidir a prioridade de direção para EI, mas em [7] foi apontado que a eficiência de MALV é melhor que a de MILV. Na realidade, nenhuma das duas afirmações é verdadeira. Neste artigo foi mostrado que para alguns objetos, normalmente os de alta textura, MILV chega a ser bem superior a MALV e para outros, normalmente os de baixa textura, o contrário se verifica.

Constatou-se que podem ocorrer variações nos desempenhos de MILV e MALV para grupos de blocos distintos, conforme o número médio de pixels do objeto. Essas variações dependem dos valores do indicador TNPO, que é uma função da forma e da textura. Para grupos com texturas muito baixas ou muito altas, a utilização do indicador TNPO, com base em limiares das texturas dos grupos de blocos, constitui uma estratégia razoável para a escolha do método mais adequado (MILV ou MALV). Contudo, à medida em que os blocos de contorno tornam-se mais preenchidos por pixels do objeto, a escolha desses limiares para a determinação da estratégia mais eficiente fica dificultada, uma vez que as variâncias dos comprimentos em ambas as direções passam a assumir valores muito próximos. Além disso, para grupos de blocos apresentando um valor de textura mediano, pode haver uma grande quantidade de blocos com baixa e alta textura, para os quais MALV e MILV, respectivamente, são mais indicados.

Portanto, foi necessário considerar uma estratégia diferente para determinar a primeira direção de processamento. Foi proposta a estratégia MACES, que mede a capacidade de compactação de energia dos coeficientes da DCT em ambas as direções. Essa estratégia mostrou-se melhor que MILV e MALV em termos de correlação entre o indicador TNPO e a determinação da primeira direção de processamento do algoritmo de extrapolação EI, à medida em que os blocos tornam-se mais cheios e apresentam, em média, valores de textura em uma faixa intermediária. Vale ressaltar que a estratégia MACES nem sempre é mais eficiente que MALV e MILV, como mostrado nos resultados apresentados neste artigo, uma vez que utiliza um bit de informação paralela para cada bloco de contorno.

Esses resultados motivaram o desenvolvimento de um método adaptativo híbrido que possibilitasse a escolha da estratégia mais indicada para cada grupo de blocos de contorno (entre MALV, MILV e MACES), utilizando o indicador TNPO como a base para o chaveamento entre as estratégias. Como a seleção da melhor estratégia pode variar com a taxa de bits, os limiares de chaveamento foram calibrados de forma a aumentar a eficiência de codificação a taxas baixas. Isso foi feito porque é nessa faixa que a EI-DCT é a estratégia mais indicada, por apresentar desempenho superior ao da SA-DCT.

O esquema híbrido proposto é comparável ou supera a eficiência da melhor entre as estratégias MILV e MALV para cada objeto. Cabe ressaltar ainda que o bloco extrapolado gerado pelo esquema proposto é independente da taxa de bits, o que representa uma vantagem sobre o esquema que escolhe como preferencial a direção que produz o menor produto  $emq \times nbits$ . Além disso, a complexidade computacional do esquema híbrido é muito menor, uma vez que não requer a extrapolação e a codificação de dois blocos para cada bloco de contorno (tomando-se ora a direção vertical como prioritária, ora a direção horizontal), e muito menos a recuperação desses blocos codificados (o que implica o processo inverso de quantização, a IDCT-2D e o método de extrapolação EI inverso) para a escolha da direção preferencial de processamento em um bloco de contorno específico.

Neste artigo foi mostrado, também, que o esquema híbrido pode ser usado em conjunto com os esquemas de partição da imagem analisados em [18]. Concluiu-se que é possível obter uma melhoria de desempenho sensível ao se usar o particionamento Flexível Simplificado, devido principalmente ao aumento da Área Efetiva dos objetos. Apesar dos esquemas de partição também alterarem o indicador TNPO, que se constitui a base do esquema híbrido, o método proposto continua sendo eficiente para a codificação dos objetos particionados com os algoritmos apontados. Os desempenhos médios de codificação a taxas baixas (que é a faixa de interesse para a utilização da EI-DCT) dos objetos utilizados nos testes apontaram que o esquema híbrido em geral é mais eficiente que MALV e MILV ao se utilizarem os particionamentos Convencional, Ortogonal Simplificado e Flexível Simplificado.

Para um conjunto de imagens paradas, a aplicação do esquema híbrido pode ser obtida de forma automática, já que a distribuição dos blocos pelos grupos necessita apenas do número de pixels do objeto de cada bloco, calculado a partir da informação de forma do objeto que é transmitida como informação paralela. Para a definição da estratégia a ser adotada para cada grupo, necessita-se antes calcular a textura do grupo, que corresponde à média das texturas dos blocos nesse grupo. Vale ressaltar, ainda, que o cálculo da textura do grupo é necessário apenas no codificador, já que o decodificador é informado de qual foi a estratégia utilizada através dos 12 bits de informação paralela. Na fase de decodificação, necessita-se apenas distribuir os blocos de contorno pelos 10 grupos. Para um sinal de vídeo, a aplicação do esquema proposto é exequível no processo de codificação desde que a definição da estratégia a ser empregada (que implica na distribuição dos blocos de contorno pelos grupos e no cálculo do indicador TNPO) seja feita quadro a quadro. Dessa forma, serão

necessários 12 bits de informação paralela para cada quadro de um objeto.

O esquema híbrido aqui proposto apresenta-se como uma opção vantajosa em relação à técnica que testa as duas direções e escolhe como preferencial aquela que produza o menor produto  $emq \times nbits$ , uma vez que os ganhos de codificação proporcionados por essa segunda técnica não compensam as desvantagens associadas à sua implementação.

## REFERÊNCIAS

- [1] F.M. Freitas and A. Alcaim, "Performance of the SA-DCT and Block-Based DCT Algorithms for Content-Based Video Coding in Terms of Morphological Features", in *Proc. 2nd International Symposium on Communications and Information Technology - ISCIT'2002*, Pattaya, Thailand, October 2002.
- [2] T. Sikora and B. Makai, "Shape-Adaptive DCT for Generic Coding of Video", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, n. 1, pp. 59-62, February 1995.
- [3] M. Bi, S.H. Ong and Y.H. Ang "Comment on Shape-Adaptive DCT for Generic Coding of Video", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, n. 6, pp. 686-688, December 1996.
- [4] G. Shen, B. Zeng and M.L. Liou, "An Efficient Hybrid Arbitrarily-Shaped Object Coding Technique", in *Proc. IEEE International Symposium on Circuits and Systems - ISCAS'2000*, Geneva, Switzerland, May 2000, p. II-5 to II-8.
- [5] E.C. Acocella and A. Alcaim, "Alignment by Phase of Vertical Coefficients in SA-DCT", *IEEE Signal Processing Letters*, vol. 8, n. 12, pp. 42-44, February 2001.
- [6] J.H. Moon, G.H. Park, S.M. Chun and S.R. Choi, "Shape-Adaptive Region Partitioning Method for Shape-Assisted Block-Based Texture Coding", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, n. 1, pp. 240-246, February 1997.
- [7] G. Shen, B. Zeng and M.L. Liou, "Achieving Optimal Rate-Distortion Performance in Arbitrarily-Shaped Transform Coding", in *Proc. IEEE International Symposium on Circuits and Systems - ISCAS'2000*, Geneva, Switzerland, May 2000, p. II-1 to II-4.
- [8] S.J. Cho, S.J. Lee, J.G. Choi and S.D. Kim, "Arbitrarily-Shaped Image Segment Coding Using Extension-Interpolation", *J. Korean Institute of Communication Science*, vol. 20, n. 9, pp. 2453-2463, 1995.
- [9] T. Xie, C.J. Weng and C.X. Feng, "A New Approach to Arbitrary Shaped DCT", in *Proc. Picture Coding Symposium - PCS'1997*, Berlin, Germany, 1997.
- [10] J.W. Yi, S.J. Cho, W.J. Kim, S.D. Kim and S.J. Lee, "A New Coding Algorithm for Arbitrarily Shaped Image Segments", *Signal Processing Image Communication*, vol. 12, pp. 231-242, 1998.
- [11] G. Shen, B. Zeng and M.L. Liou, "A New Padding Technique for Coding of Arbitrarily-Shaped Image/Video Segments", in *Proc. IEEE International Conference on Image Processing - ICIP'1999*, Kobe, Japan, October 1999.
- [12] A. Kaup, "Object-Based Texture Coding of Moving Video in MPEG-4", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, n. 1, pp. February 1999.
- [13] Y. Luo, and R.W. Ward, "Removing the Blocking Artifacts of Block-Based DCT Compressed Images", *IEEE Transactions on Image Processing*, vol. 12, n. 7, pp. 838-842, July 2003.
- [14] F.M. Freitas e A. Alcaim, "Escolha da Direção Preferencial de Processamento do Algoritmo de Extrapolação EI para Codificação de Imagens Baseada em Objeto", *Anais do 21o. Simpósio Brasileiro de Telecomunicações - SBT'2004*, Belém, PA, Setembro 2004.
- [15] R.C. Reininger, J.D. Gibson, "Distribution of the Two-Dimensional DCT Coefficients for Images", *IEEE Transactions on Communications*, vol. 31, no. 6, pp. 835-839, June 1983.
- [16] N.S. Jayant, P. Noll, "Digital Coding of Waveforms - Principles and Applications to Speech and Video", *Prentice Hall Signal Processing Series*, 1984.
- [17] A.K. Jain, "Fundamentals of Digital Image Processing", *Prentice Hall Information and System Sciences Series*, 1989.
- [18] F.M. Freitas e A. Alcaim, "Análise de Métodos de Partição em Blocos em Codificação de Vídeo Baseada em Objeto", *Anais do 18o. Simpósio Brasileiro de Telecomunicações - SBT'2001*, Fortaleza, CE, Setembro 2001.

**Flávia Magalhães Freitas** graduou-se em Engenharia Elétrica com ênfase em sistemas eletrônicos pela Pontifícia Universidade Católica de Minas Gerais em 1990, obteve o grau de Mestre em Engenharia Elétrica pela Universidade Federal de Minas Gerais em 1995 e o grau de Doutor em Engenharia Elétrica pela Pontifícia Universidade Católica do Rio de Janeiro, em 2004. Desde 1991 é professora do Instituto Politécnico da Pontifícia Universidade Católica de Minas Gerais, onde ocupa o cargo de Professor Adjunto III. Ela foi Coordenadora de Pesquisa dessa universidade no período de 1997 a 2003 e atualmente integra o corpo docente do Programa de Pós-graduação em Engenharia Elétrica.

**Abraham Alcaim** recebeu o diploma de Engenheiro Eletricista e o título de Mestre em Ciências em Engenharia Elétrica pela Pontifícia Universidade Católica do Rio de Janeiro (PUC/Rio) em 1975 e 1977, respectivamente, e os títulos de D.I.C. e Ph.D. pelo Imperial College of Science and Technology, University of London, em 1981. Desde 1976 ele é professor do Centro de Estudos em Telecomunicações da Universidade Católica (CETUC), tendo atualmente o cargo de Professor Associado. O Dr. Alcaim trabalha há mais de 25 anos nas áreas de processamento digital de voz e imagem. Ele é autor de diversos artigos publicados em congressos e revistas nacionais e internacionais. Em 1984 ele esteve por um período curto com o Centre National d'Etudes des Télécommunications (CNET), em Lannion, França, onde trabalhou em medidas de qualidade objetivas e subjetivas para codificadores de voz. De dezembro de 1991 a setembro de 1993 ele foi Cientista Visitante no Centro Científico Rio da IBM Brasil, onde trabalhou no projeto de novos codificadores de imagem, com aplicação especial para imagens obtidas por satélites de sensoriamento remoto. O Dr. Alcaim foi o Technical Program Chairman dos simpósios internacionais SBT/IEEE International Telecommunications Symposium de 1990 e 1994, e o Executive Chairman da IEEE Global Telecommunications Conference de 1999. Ele foi membro do Conselho Deliberativo da SBrT no período de 1996 a 2001 e membro do Comitê de Assessoramento de Engenharia Elétrica, Biomédica e Microeletrônica (CA-EE) do CNPq no período de 1998 a 2001. De janeiro de 2001 a dezembro de 2004 o Dr. Alcaim foi Editor da área de Processamento de Sinais da Revista da SBrT.