

## 5

# Escolha da Direção Preferencial de Processamento da EI-DCT

### 5.1

#### 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. Conforme apresentado em capítulos anteriores, há basicamente duas abordagens para processar a imagem: a primeira usa a DCT adaptativa à forma (SA-DCT) [67]-[77] e a segunda emprega esquemas de DCT baseada em blocos após processar os blocos de contorno usando-se algoritmos de extrapolação [76]-[83]. Neste capítulo serão propostas novas estratégias que visam à melhoria de desempenho da DCT baseada em blocos associada ao algoritmo de extrapolação EI (*Extension/Interpolation*) - EI-DCT. Como foi visto no Capítulo 4, a EI-DCT tem se mostrado uma das técnicas mais eficientes a taxas baixas e médias. O algoritmo de extrapolação EI aplica um conjunto de transformadas 1D, seguidas de inserção de zeros no domínio da transformada e um outro conjunto de transformadas 1D [78],[80].

A direção sobre a qual se aplica o primeiro conjunto de transformadas 1D no algoritmo de extrapolação EI influencia o desempenho final do codificador. Em [77] 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. Obviamente, o critério supera em desempenho outras estratégias existentes para a escolha da direção preferencial de processamento. Contudo, ele apresenta a desvantagem de uma elevadíssima carga computacional, além de

exigir o cálculo de um novo conjunto de coeficientes a cada vez que ocorrer mudança na taxa de transmissão.

Outros esquemas para a determinação da primeira direção de processamento do algoritmo EI baseiam-se exclusivamente na forma do objeto segmentado [77],[80]. Sendo assim, não necessitam transmitir nenhuma informação paralela adicional. 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) [80], que escolhe a direção associada à menor das variâncias e (2) MALV (MAXimum Lengths Variances) [77], que seleciona a direção associada à maior variância.

Na literatura ainda não há um consenso 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.

Neste capítulo, busca-se inicialmente avaliar essas conclusões contraditórias através de uma análise criteriosa dessas estratégias, com base em indicadores que relacionem características morfológicas e de textura. Em seguida é proposto um esquema híbrido que seleciona adaptativamente a melhor estratégia para decidir quanto à prioridade da direção vertical ou horizontal para o algoritmo EI. Essa decisão é tomada com base nos indicadores morfológicos propostos no Capítulo 4 para análise de codificadores orientados por objeto. Finalmente, será avaliada a influência dos algoritmos de partição da imagem (vide Capítulo 3) sobre o desempenho final do esquema híbrido.

A Seção 5.2 apresenta uma breve descrição do algoritmo EI e alguns resultados preliminares relativos aos desempenhos das estratégias MALV e MILV segundo o indicador morfológico TNPO. Para os casos em que é difícil estabelecer uma correlação entre esse indicador e as estratégias baseadas nas variâncias dos comprimentos, propõe-se uma estratégia alternativa, chamada MACES, que mede a capacidade de compactação de energia das transformadas. Essa estratégia está descrita na Seção 5.3, onde também são apresentadas comparações de desempenho entre MILV, MALV e MACES. O esquema híbrido proposto, baseado nos indicadores morfológicos apresentados no Capítulo 4, é descrito na seção 5.4. A Seção 5.5 contém resultados

de simulação, com discussões. Na Seção 5.6, avalia-se a possibilidade da utilização do esquema híbrido associado aos algoritmos de partição da imagem que tenham sido apontados como os mais eficientes no Capítulo 3. Finalmente, a Seção 5.7 é dedicada às conclusões do capítulo.

## 5.2

### Resultados preliminares das Estratégias MILV e MALV Aplicadas ao Algoritmo EI

Originalmente, a direção vertical foi a utilizada como preferencial no algoritmo de extrapolação EI [78]. Posteriormente, em [80] a estratégia MILV foi empregada para a escolha da primeira direção de processamento (direção preferencial) nesse algoritmo. Por outro lado, em [77], em oposição ao que foi considerado em [80], a estratégia MALV foi apresentada como uma melhor opção para a escolha da direção preferencial. Entretanto, nossos resultados experimentais preliminares indicaram que nenhuma das duas técnicas pode ser sempre considerada como a melhor estratégia. Para alguns objetos, normalmente aqueles de baixa textura, nós observamos que MALV é superior à MILV e para outros, normalmente os de alta textura, o contrário se verifica.

A escolha da direção mais adequada em um determinado grupo de blocos de contorno depende de características morfológicas e também da textura. Um indicador que incorpora essas características é o TNPO (Texturas de acordo com o Número de Pixels do Objeto), descrito na Seção 4.3.

Conforme foi visto no Capítulo 4, para a obtenção do indicador TNPO, os blocos de contorno são classificados em 10 grupos, G1 a G10, e o grupo G11 é representado por todos os blocos cheios. O desvio padrão das luminâncias dos pixels em cada bloco é a textura do bloco e o valor médio das texturas dos blocos pertencentes a um determinado grupo representa a textura do grupo, que é armazenada no indicador TNPO.

O que se pretende verificar inicialmente é se pode ser feita a opção por MILV ou MALV em um grupo de blocos específico, com boa precisão, com base no indicador TNPO. Ou seja, observando-se quão cheio ou vazio de pixels do objeto ele é, bem como a textura média nos blocos com um número de pixels equivalente. A fim de facilitar o acompanhamento do texto, o indicador TNPO para os objetos descritos no Capítulo 4 foi repetido na Figura 5.1. Observa-se que os objetos *AKYIO* e *PEPPER* apresentam baixas texturas para todos os grupos, ao passo que *CHILDREN* e *FISH*

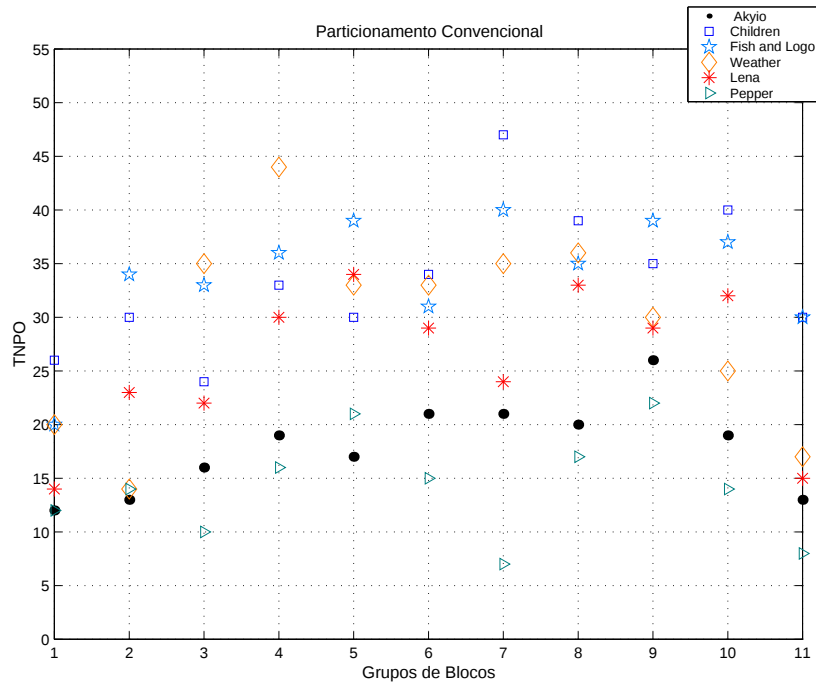


Figura 5.1: Textura de acordo com o Número de Pixels do Objeto - TNPO

*AND LOGO* normalmente apresentam texturas mais elevadas. *LENA* e *WEATHER* apresentam, em geral, texturas de médias a altas, dependendo do grupo.

Nas Figuras 5.2 a 5.7 são apresentados os resultados de codificação dos objetos, ao se usar as estratégias MALV e MILV associadas ao algoritmo EI-DCT. Apesar das duas estratégias produzirem matrizes de alocação de bits ligeiramente diferentes, optou-se pela visualização somente dos blocos de contorno porque o desempenho dos blocos inteiros será praticamente igual em ambos os esquemas (tratam-se ambos da abordagem DCT baseada em blocos). Nessas figuras, observa-se uma tendência: a baixas taxas de bits, aproximadamente até 2 bppo, a estratégia MILV geralmente é mais eficiente para todos os objetos. A partir dessa taxa, normalmente os objetos em que todos os grupos têm textura baixa, como *AKYIO* e *PEPPER*, são melhor codificados se a direção preferencial para a extrapolação dos blocos de contorno for indicada pela estratégia MALV, ao passo que os objetos que apresentam textura elevada em praticamente todos os grupos, como *CHILDREN* e *FISH AND LOGO*, são mais eficientemente codificados se os blocos de contorno forem processados primeiro pela direção apontada pela estratégia MILV.

Esses resultados mostram que a afirmação em [77] de que MALV é melhor que MILV, ao se apresentar o desempenho da codificação de apenas um quadro da sequência *WEATHER*, não é verdadeira. Basta observar a

Figura 5.2, que mostra os resultados de codificação para os 11 primeiros quadros da sequência *WEATHER*, onde se observa que MALV foi superior a MILV somente a taxas de médias a altas. Para taxas baixas, embora ambas as estratégias proporcionem desempenhos praticamente idênticos, ocorreu uma ligeira supremacia de MILV em relação a MALV.

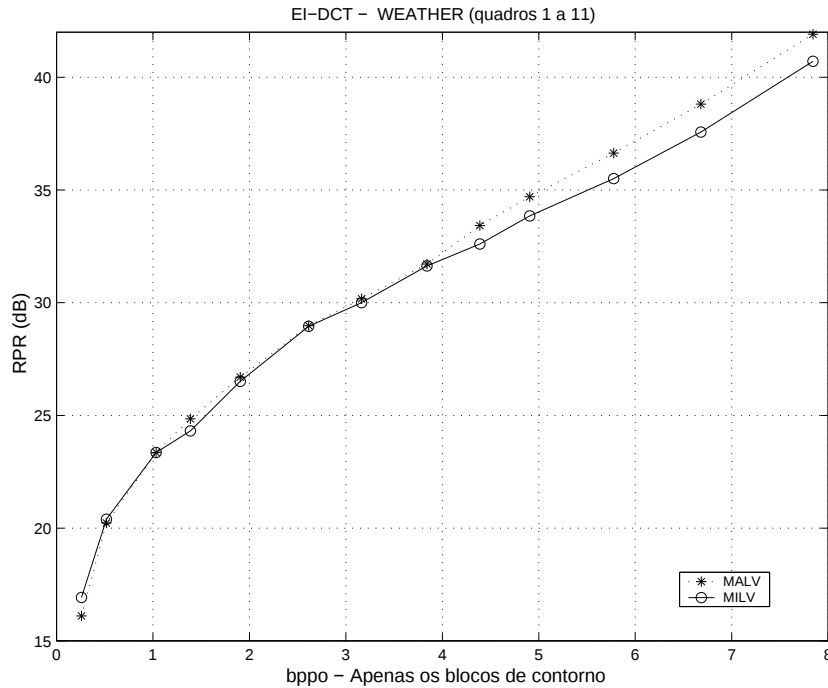


Figura 5.2: Resultados de codificação com a EI-DCT para 11 quadros do objeto da sequência *WEATHER*

Dos resultados experimentais nós verificamos, ainda, que as eficiências das estratégias MILV e MALV apresentam uma variação não apenas de objeto para objeto, como também, em um mesmo objeto, de grupo para grupo de blocos. 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). Já nos grupos com texturas médias, onde pode haver um número significativo de blocos com alta textura e também outros com baixa textura, a análise com base unicamente no indicador TNPO é dificultada. Observamos também que o limiar das texturas classificadas como altas ou baixas variam de grupo para grupo. Nos blocos de contorno medianamente ou bastante preenchidos por pixels do objeto, torna-se mais difícil a escolha desses limiares para a determinação da estratégia mais eficiente.

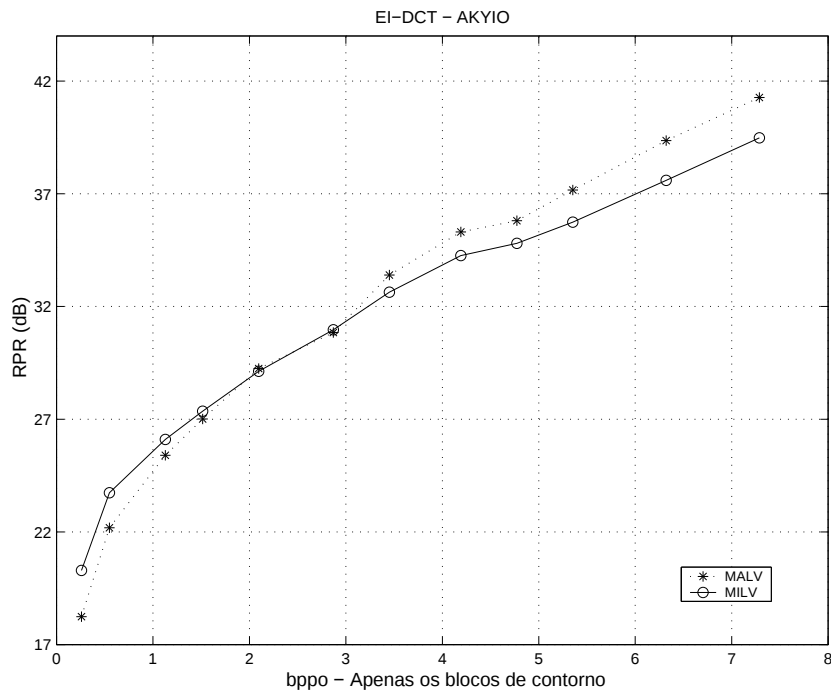


Figura 5.3: Resultados de codificação com a EI-DCT para o objeto do quadro *AKYIO*

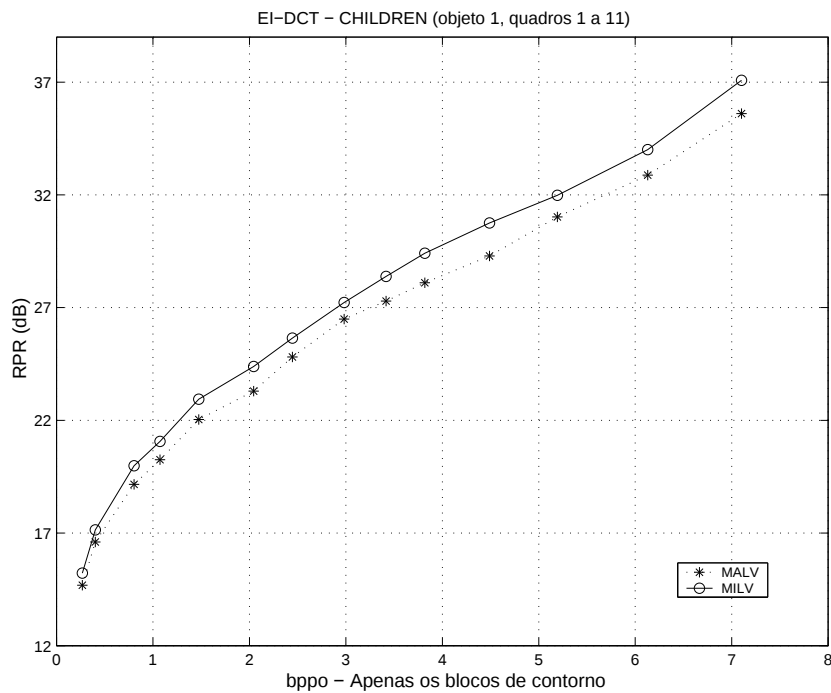


Figura 5.4: Resultados de codificação com a EI-DCT para 11 quadros de um objeto da sequência *CHILDREN*

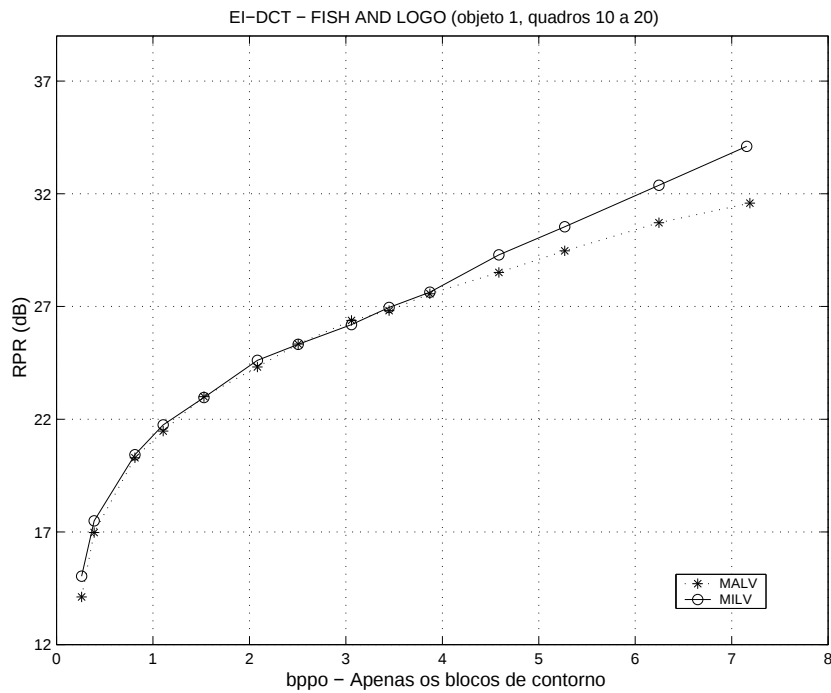


Figura 5.5: Resultados de codificação com a EI-DCT para 11 quadros de um objeto da sequência *FISH AND LOGO*

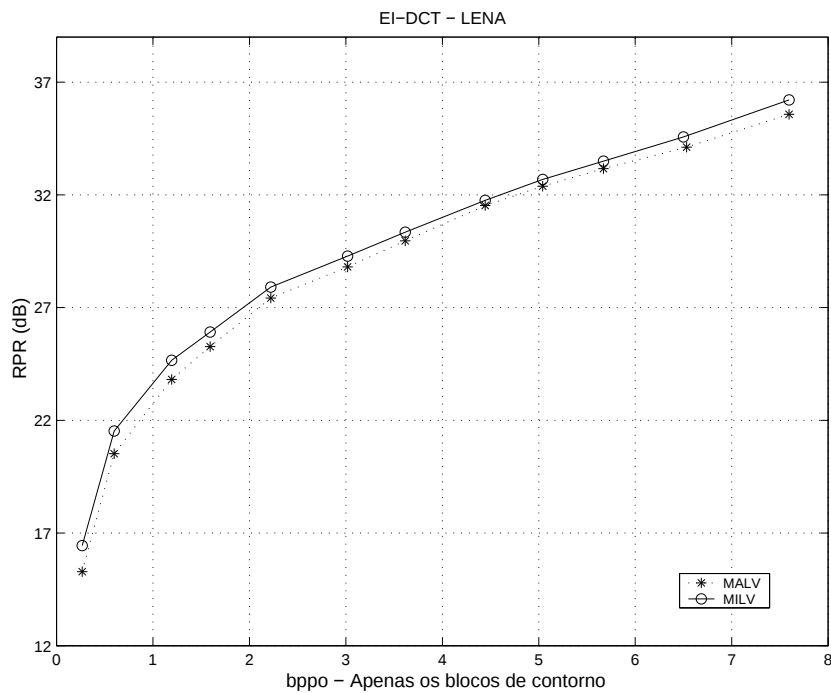


Figura 5.6: Resultados de codificação com a EI-DCT para o objeto do quadro *LENA*

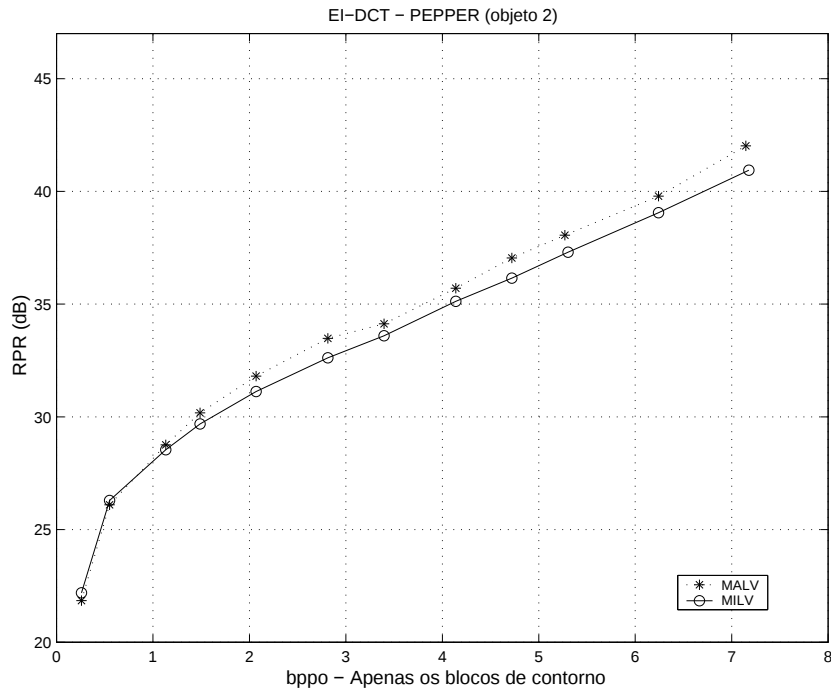


Figura 5.7: Resultados de codificação com a EI-DCT para um objeto do quadro *PEPPER*

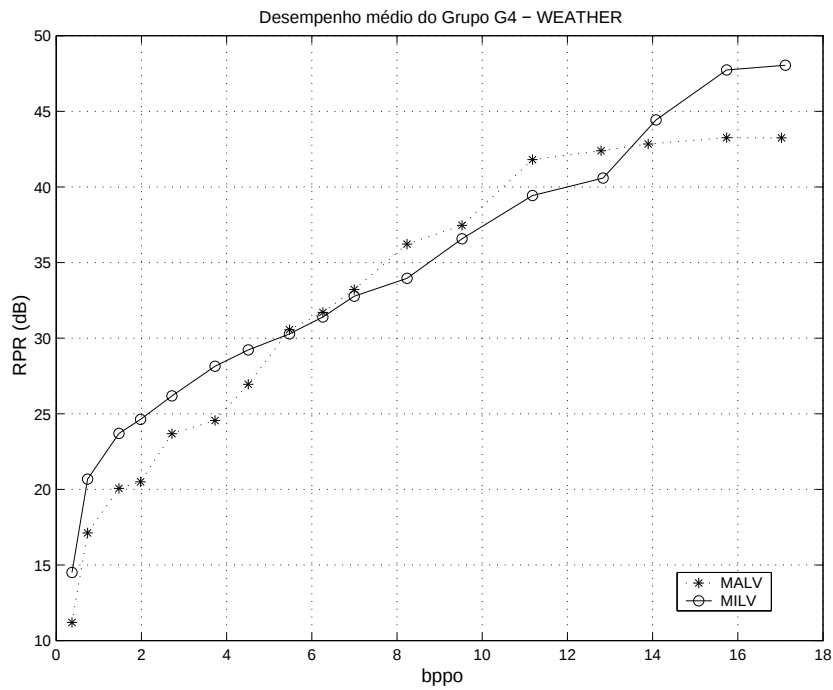


Figura 5.8: Resultados de codificação dos blocos do grupo G4 de *WEATHER* (alta textura)

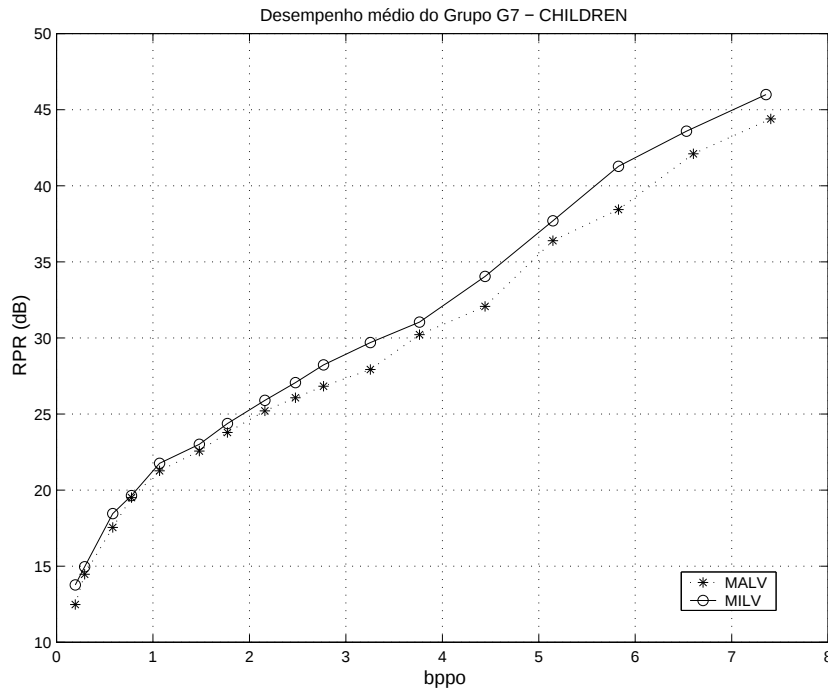


Figura 5.9: Resultados de codificação dos blocos do grupo G7 de *CHILDREN* (alta textura)

Como será observado nas figuras seguintes, a melhor adequação de MILV ou MALV, em um grupo de blocos específico, pode variar com a taxa de bits em *bppo*. Procuraremos relacionar o indicador TNPO ao desempenho de MILV e MALV a taxas baixas, para as quais a abordagem de DCT baseada em blocos é reconhecida na literatura como a mais indicada. Para essas taxas, é possível apresentar alguns argumentos que justifiquem a superioridade de MALV ou MILV, em função da textura. É importante ressaltar que a condição de ‘taxas baixas’, caso a taxa usada como parâmetro seja a taxa de *bppo*, varia de grupo para grupo. Os grupos de blocos muito vazios apresentarão, para uma dada taxa média de *bppo* requerida para a codificação do objeto completo, uma altíssima taxa de *bppo* e à medida em que os blocos tornam-se mais cheios, a taxa de *bppo* correspondente vai se tornando menor, já que todos os blocos utilizam a mesma quantidade de bits para a sua codificação.

Na estratégia MALV, a direção preferencial será aquela onde a maioria dos pixels do objeto está localizada em segmentos cheios, uma vez que uma grande variância dos comprimentos está associada à existência de segmentos muito cheios e muito vazios. Portanto, em blocos de baixa textura, onde todos os segmentos tendem a estar mais suavizados mesmo antes da extração, é melhor usar a estratégia MALV, que aplica a primeira DCT-1D sobre segmentos maiores. Com uma quantidade restrita de bits (taxas bai-

xas), a maioria dos pixels, que estará localizada nesses segmentos longos, será eficientemente codificada. Por outro lado, na estratégia MILV, a direção preferencial será aquela onde os pixels estão distribuídos por segmentos de tamanhos mais uniformes, já que uma pequena variância dos comprimentos está associada à existência de segmentos com comprimentos próximos ao valor médio. Isso significa que em blocos de alta textura, após a primeira DCT-1D, esses segmentos serão mais suavizados que os segmentos longos de MALV, proporcionando a compactação de energia em um menor número de coeficientes. A justificativa para isso é que, de acordo com o algoritmo de extrapolação EI, um segmento de tamanho  $L$ , onde  $1 \leq L < N$ , é transformado aplicando-se uma DCT-1D de dimensão  $L$ , em seguida são inseridos  $(N - L)$  zeros no domínio da transformada e finalmente, aos  $N$  coeficientes resultantes aplica-se a IDCT-1D de dimensão  $N$ , produzindo um conjunto de  $N$  pixels interpolados. Portanto, quanto menor o comprimento original  $L$ , mais a textura será suavizada no segmento interpolado de tamanho  $N$ , uma vez que a energia do segmento original será distribuída pelos  $N$  pixels interpolados. Esse fator também será útil às aplicações a taxas baixas. Como exemplo são mostradas as Figuras 5.8 e 5.9, que ilustram casos onde o indicador TNPO apresenta valores altos: grupo  $G4$  de *WEATHER* e grupo  $G7$  de *CHILDREN* (vide Figura 5.1). Note-se que no primeiro caso, Figura 5.8, a estratégia MILV foi mais indicada a taxas baixas, para as quais a EI-DCT é a mais indicada, por apresentar a maior eficiência. Já no segundo caso, Figura 5.9, a estratégia MILV foi a mais eficiente, independente da taxa usada.

Nos grupos  $G1$  (blocos muito vazios) e  $G10$  (blocos muito cheios), as estratégias MALV e MILV fornecem desempenhos razoavelmente equivalentes para todos os objetos. As diferenças maiores ocorreram nos blocos medianamente preenchidos. No grupo  $G2$ , por exemplo, observou-se uma forte dependência entre o indicador TNPO e a eficiência das estratégias MILV e MALV. As Figuras 5.10 a 5.15 apresentam o desempenho de codificação dos blocos do grupo  $G2$  para os objetos examinados. Em *AKYIO* (Figura 5.10) e *LENA* (Figura 5.11), onde os blocos do grupo  $G2$  apresentam texturas de valores baixos, a estratégia MALV foi superior à estratégia MILV. Note-se que para esses objetos, dependendo da taxa, pode-se ter diferenças em RPR de mais de 7 dB. Por outro lado, em *CHILDREN* (Figura 5.13) e *FISH AND LOGO* (Figura 5.14), onde os blocos do grupo  $G2$  apresentam texturas mais altas, a estratégia MILV foi superior. Vale ressaltar que, dependendo da taxa, as diferenças em RPR chegam a cerca de 7 dB no caso de *FISH AND LOGO*. Já para os blocos do grupo  $G2$  de *WEATHER* e *PEPPER*,

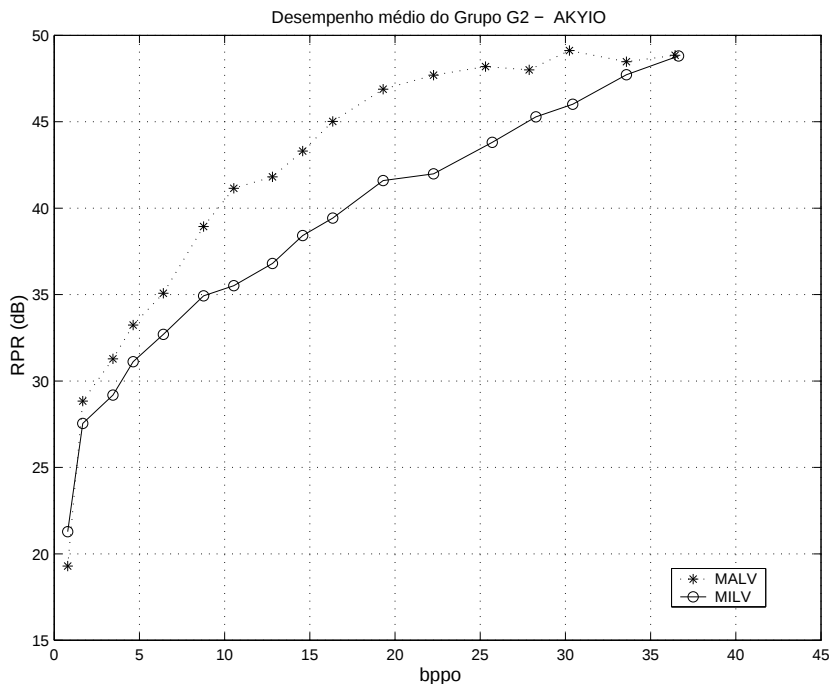


Figura 5.10: Resultados de codificação dos blocos do grupo G2 de *AKYIO*

que também têm textura baixa, os desempenhos de MILV e MALV se alternam, como mostram as Figuras 5.15 e 5.12. Entretanto, as diferenças de desempenho em geral são pequenas.

À medida em que os blocos tornam-se mais cheios, as variâncias dos comprimentos dos segmentos em cada uma das direções tendem a assumir valores mais próximos e o parâmetro TNPO deixa de ser um bom indicador para a escolha entre MILV e MALV, principalmente nos grupos cuja textura assume um valor mediano (entre 25 a 35, aproximadamente). Nesses grupos, pode haver um número razoável de blocos com baixa textura e outros com alta textura, que desempenham melhor com MALV e MILV, respectivamente. De *G6* a *G10*, por exemplo, a estratégia MALV só pode ser apontada como a mais eficiente nos casos de textura extremamente baixa (abaixo de 25, aproximadamente), ao passo que a estratégia MILV é a mais indicada nos casos de textura extremamente alta (superior a 35).

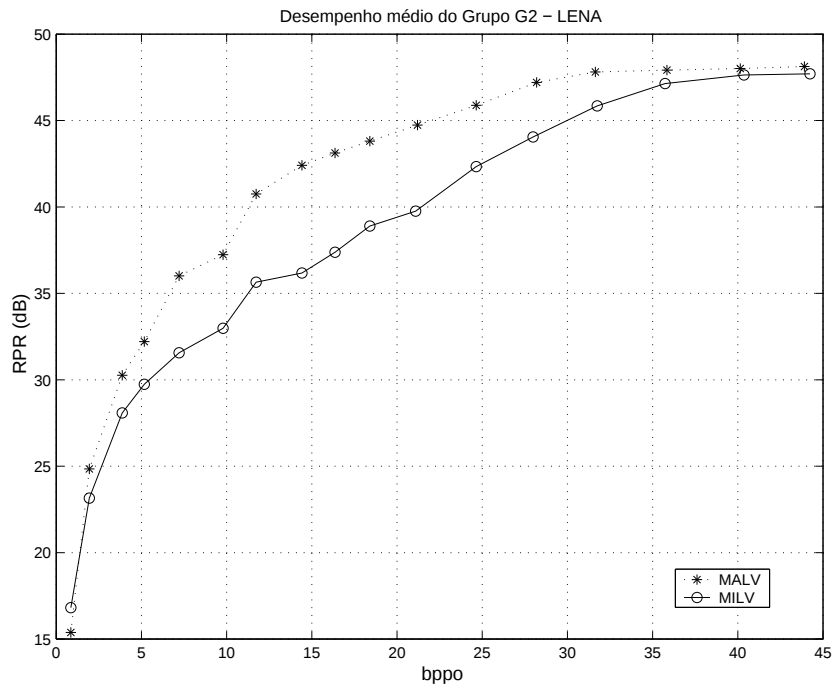


Figura 5.11: Resultados de codificação dos blocos do grupo G2 de *LENA*

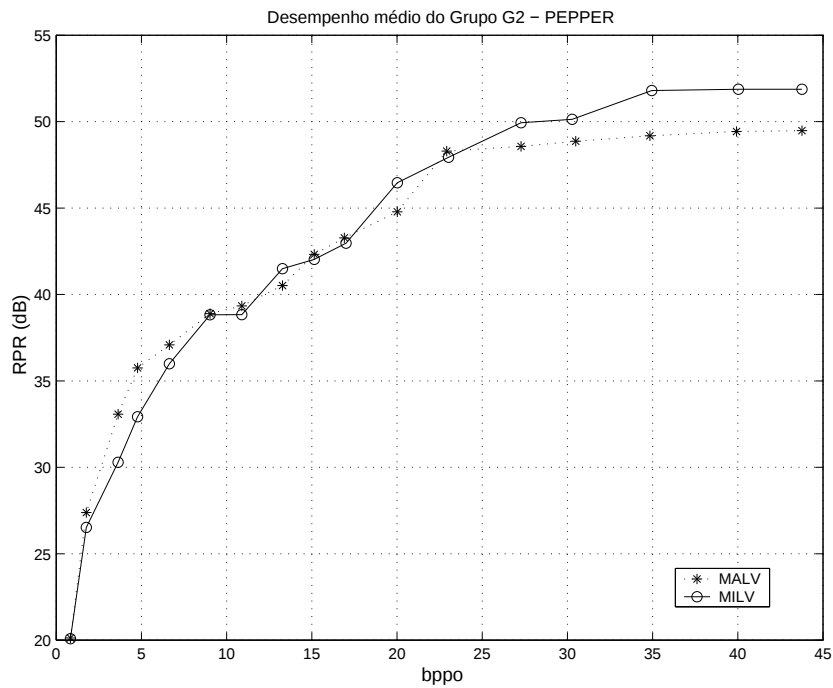


Figura 5.12: Resultados de codificação dos blocos do grupo G2 de *PEPPER*

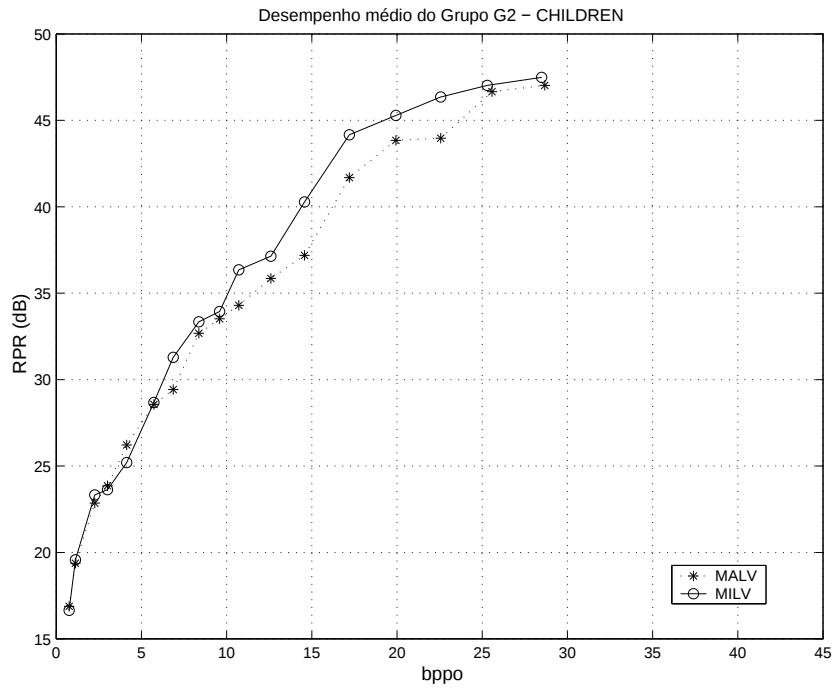


Figura 5.13: Resultados de codificação dos blocos do grupo G2 de *CHILDREN*

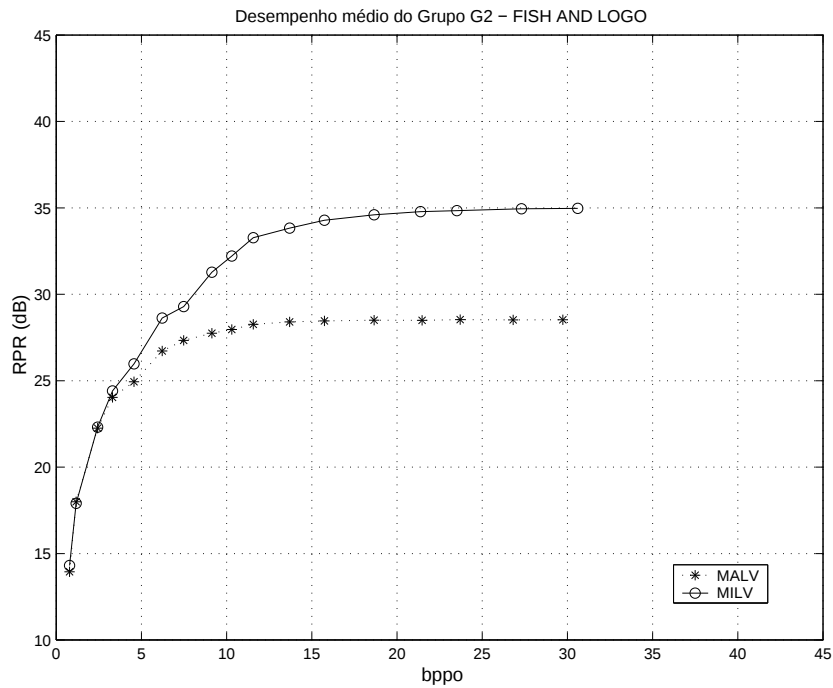


Figura 5.14: Resultados de codificação dos blocos do grupo G2 de *FISH AND LOGO*

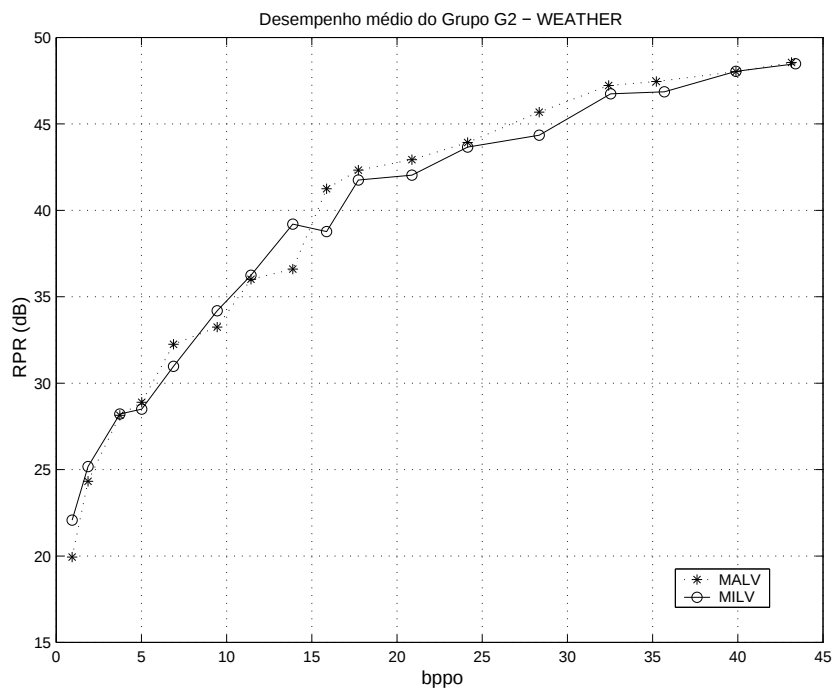


Figura 5.15: Resultados de codificação dos blocos do grupo G2 de *WEATHER*

### 5.3

#### A Estratégia MACES

A análise apresentada na seção anterior consistia em 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 nenhuma 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 ambas as direções. Após uma investigação experimental cuidadosa de diversas medidas de compactação de energia nos casos em que MILV e MALV não podem ser usados eficientemente, foi proposta a estratégia *MAximum Cumulative Energy Sum* (MACES) como medida de desempenho nessas situações. Como será visto em seguida, 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 no próximo parágrafo, 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 é dada por:

$$CE(k) = \sum_{m=0}^k [c(m)]^2 \quad (5-1)$$

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. A comparação entre ambos os vetores foi realizada para determinados valores de  $k$ , correspondendo à energia cumulativa nos 25%, 50%, 75% e finalmente, 100% dos coeficientes de maior amplitude.

Contudo, foi observado que uma estratégia que permitisse avaliar qual dos dois vetores apresentasse, em média, os maiores valores, seria mais adequada. Sendo assim, para a determinação da direção preferencial de processamento, foi proposta a utilização da soma das componentes do vetor  $CE$  para  $k = 0, \dots, H$ . Uma 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 dos 32 coeficientes de maior magnitude é computada, para cada uma das duas direções, por:

$$CES = \sum_{k=0}^{31} CE(k) \quad (5-2)$$

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, o bloco extrapolado é independente da taxa de bits e, para a escolha da direção diferencial, o algoritmo não requer a recuperação do bloco codificado, o que representam importantes vantagens sobre o critério que escolhe a direção que produz o menor produto  $emq \times nbits$ .

Os resultados de simulações mostradas nas Figuras 5.16 a 5.26, comparam, para alguns grupos de blocos, as eficiências das estratégias MILV, MALV e MACES. As Figuras 5.16 a 5.18 mostram o desempenho de codificação do grupo G2, onde 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 5.19 a 5.21 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.

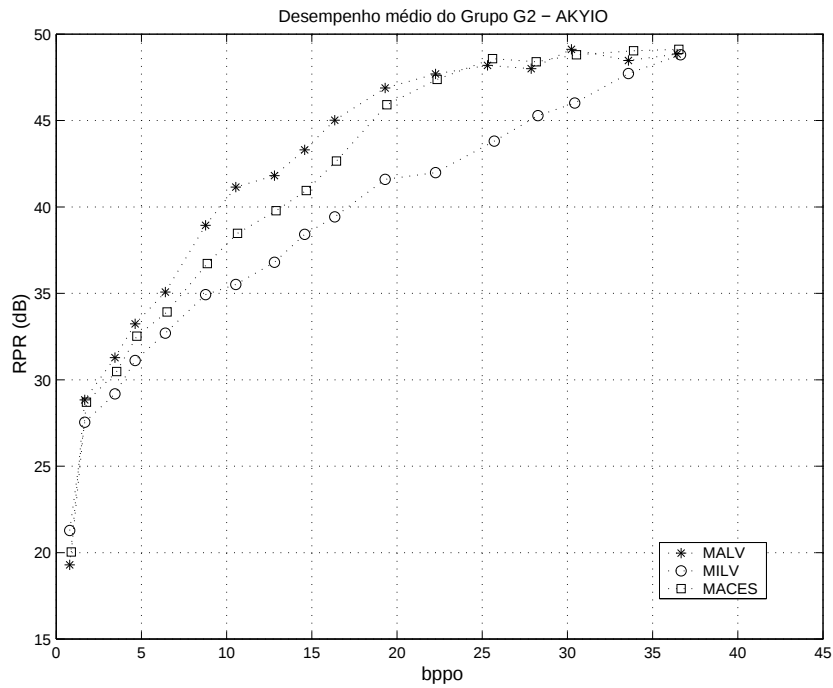


Figura 5.16: Resultados de codificação dos blocos do grupo G2 de *AKYIO*

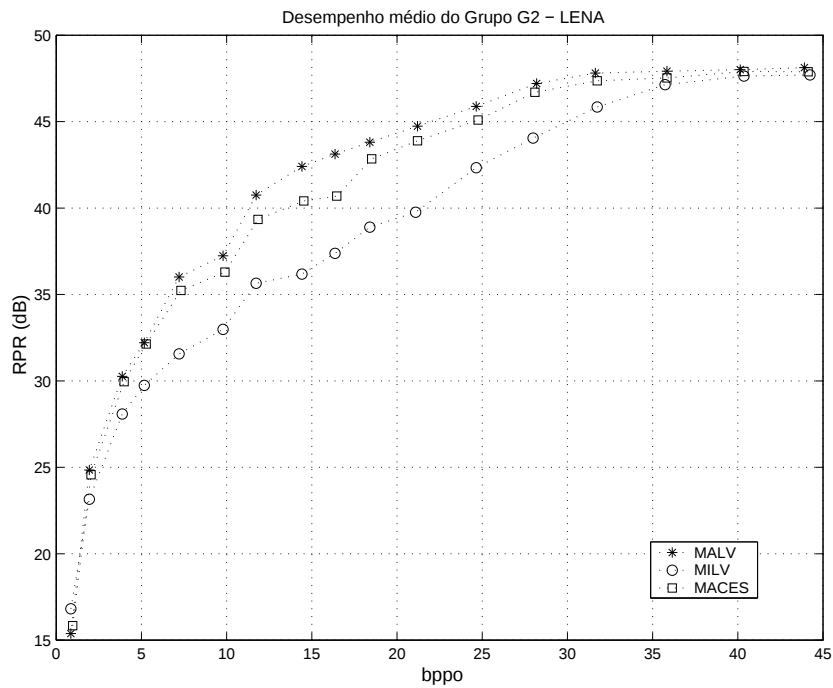


Figura 5.17: Resultados de codificação dos blocos do grupo G2 de *LENA*

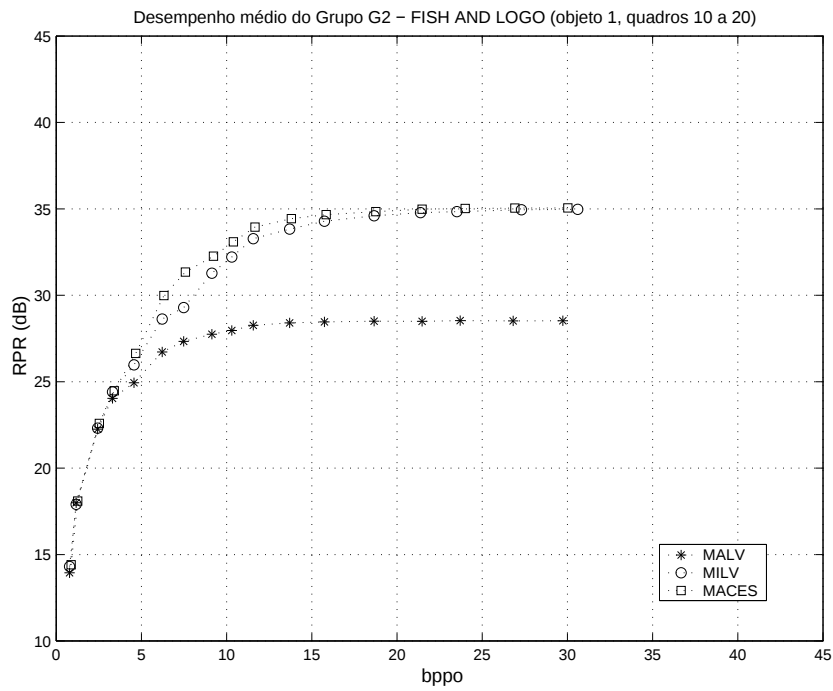


Figura 5.18: Resultados de codificação dos blocos do grupo G2 de *FISH AND LOGO* (objeto 1, quadros 10 a 20)

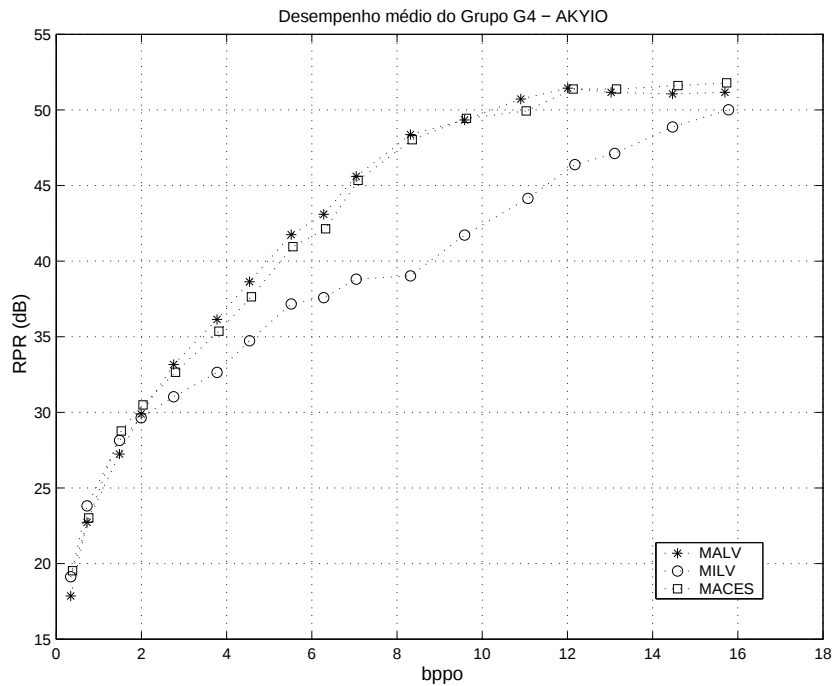


Figura 5.19: Resultados de codificação dos blocos do grupo G4 de *AKYIO*

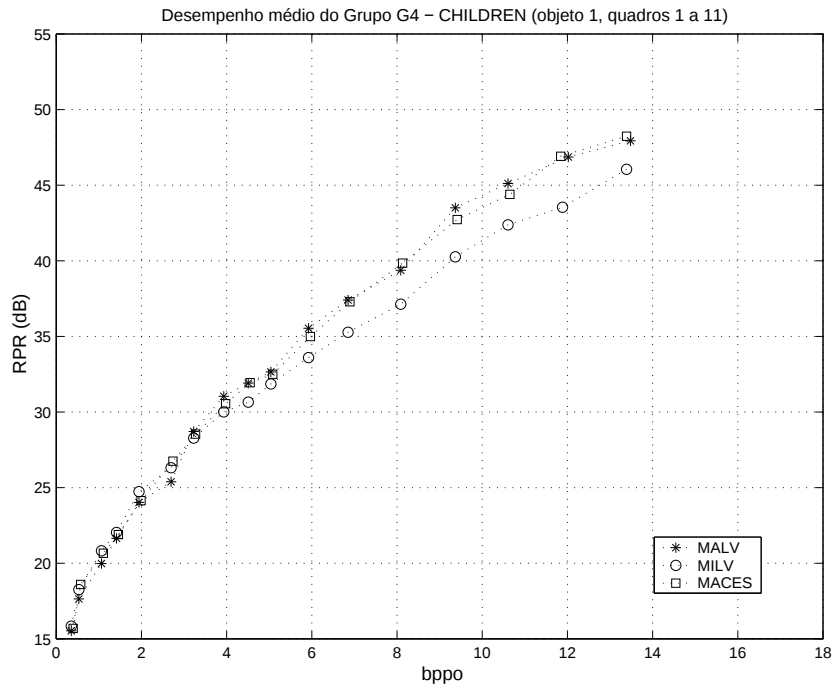


Figura 5.20: Resultados de codificação dos blocos do grupo G4 de *CHILDREN* (objeto 1, quadros 1 a 11)

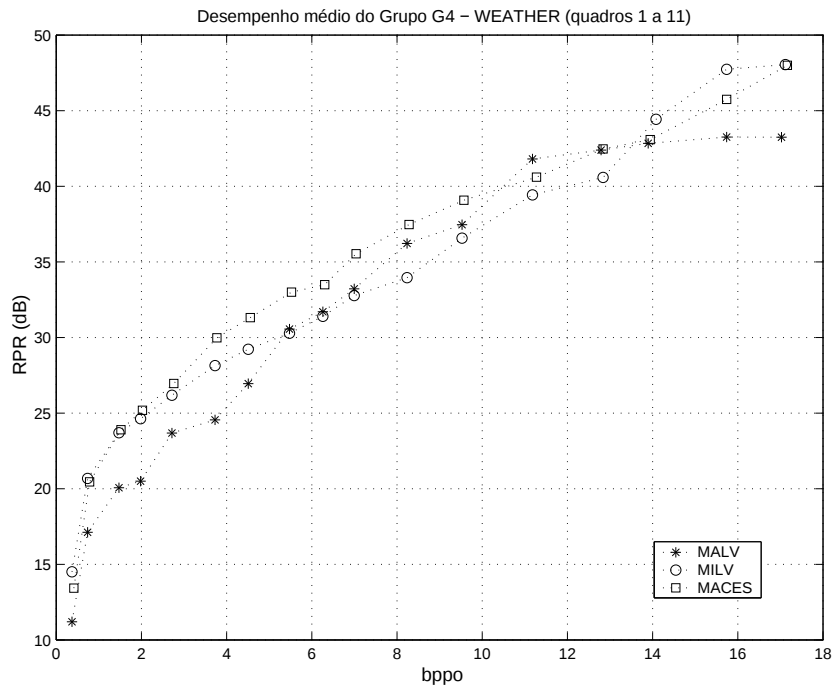


Figura 5.21: Resultados de codificação dos blocos do grupo G4 de *WEATHER* (quadros 1 a 11)

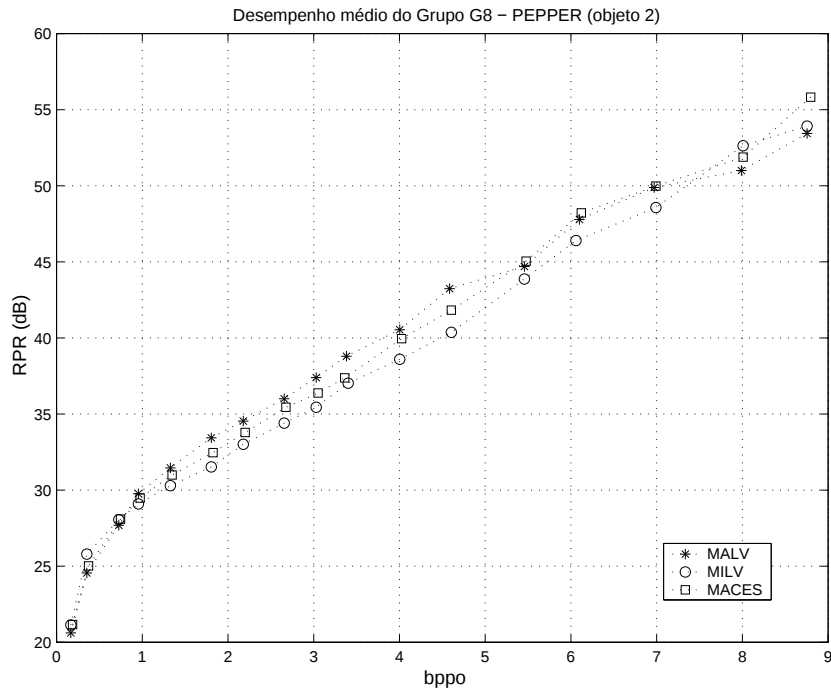


Figura 5.22: Resultados de codificação dos blocos do grupo G8 de *PEPPER* (objeto 2)

Quando os blocos estão mais cheios, torna-se cada vez mais difícil apontar, segundo o indicador TNPO, qual a 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 (onde normalmente MALV é melhor) ou muito alta (onde MILV é melhor). As Figuras 5.22 a 5.24 ilustram situações de textura muito baixa (*PEPPER*), média (*FISH AND LOGO*) e muito alta (*CHILDREN*) em *G8*, para os 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 5.25 e 5.26, 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.

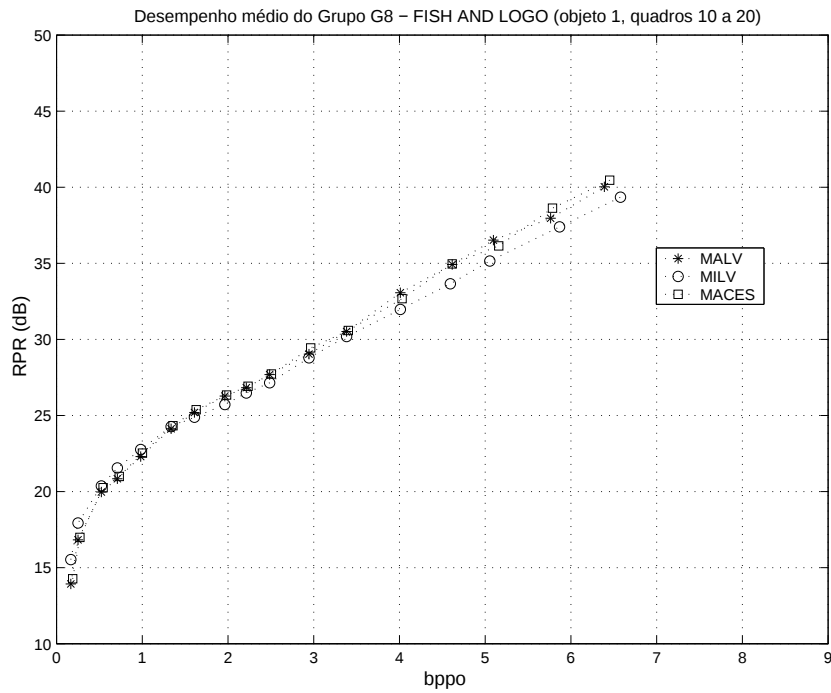


Figura 5.23: Resultados de codificação dos blocos do grupo G8 de *FISH AND LOGO* (objeto 1, quadros 10 a 20)

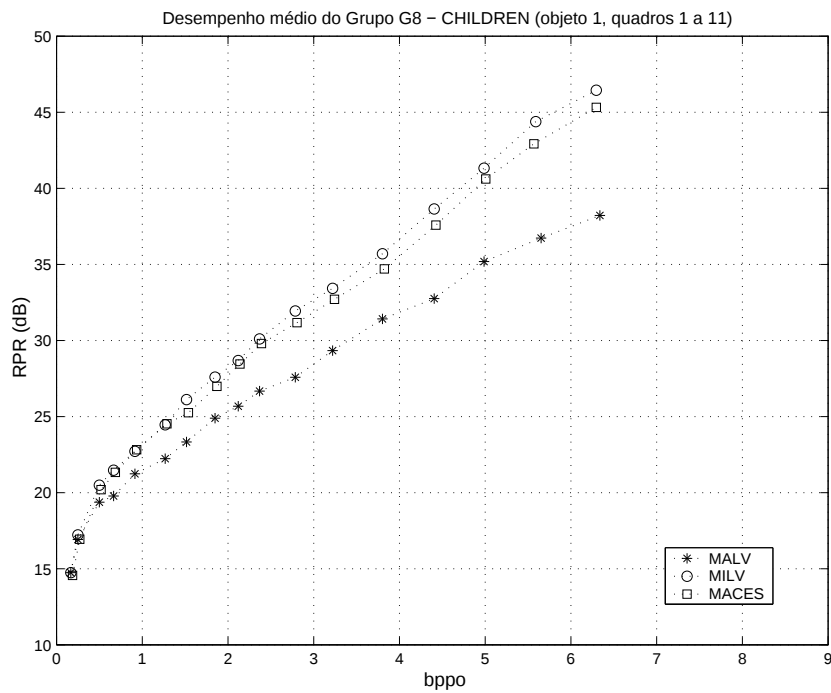


Figura 5.24: Resultados de codificação dos blocos do grupo G8 de *CHILDREN* (objeto 1, quadros 1 a 11)

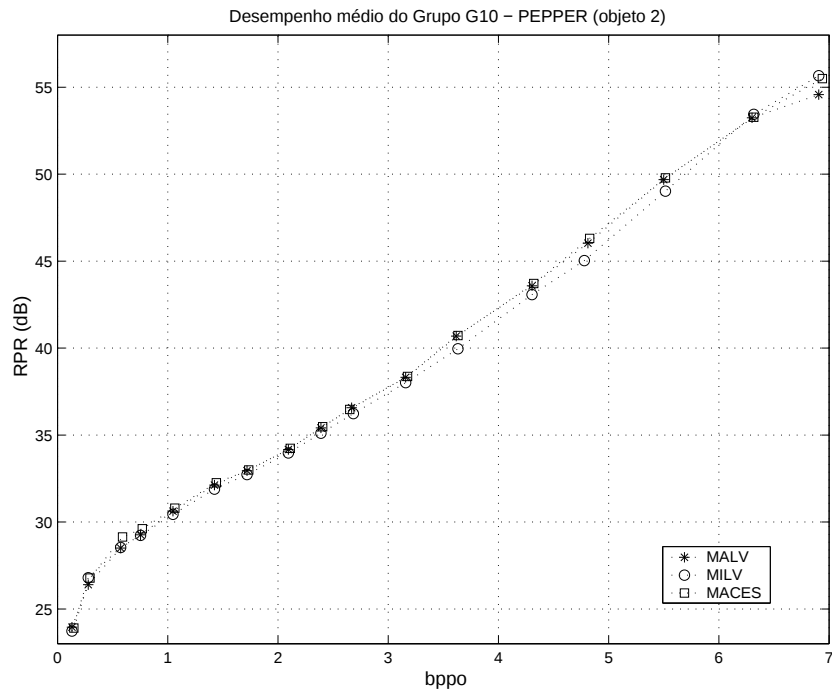


Figura 5.25: Resultados de codificação dos blocos do grupo G10 de *PEPPER* (objeto 2)

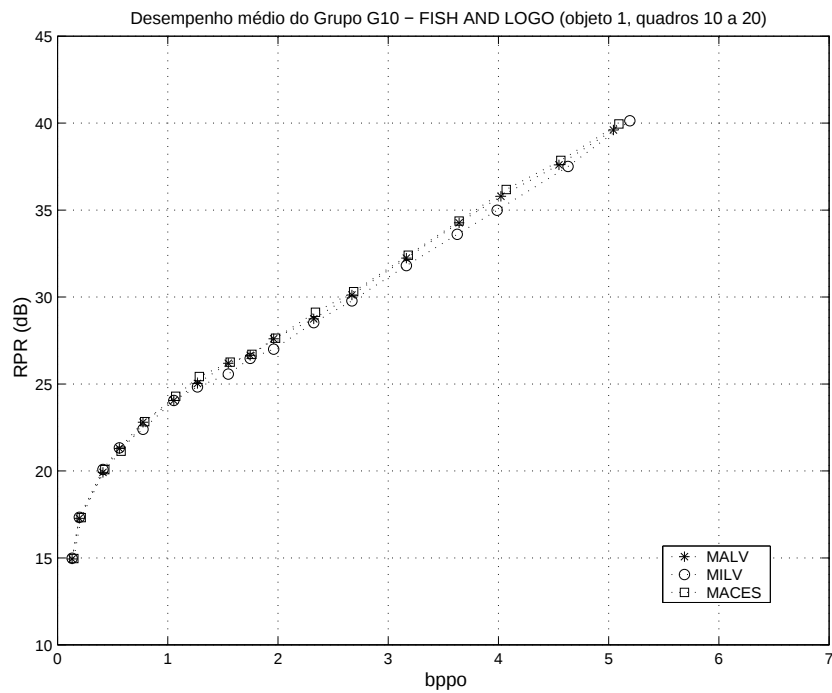


Figura 5.26: Resultados de codificação dos blocos do grupo G10 de *FISH AND LOGO* (objeto 1, quadros 10 a 20)

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

- MALV é mais indicada em grupos de baixa textura e 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 ( $G10$ ), onde praticamente não existe diferença entre as variâncias dos comprimentos nas duas direções.

## 5.4

### O Esquema Híbrido Proposto para a EI-DCT

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 MILV, MALV ou MACES. Baseando-se no indicador TNPO, pode-se selecionar a estratégia mais adequada para cada grupo  $g \in \{G1, \dots, G10\}$ . Estudos experimentais levaram aos limiares de textura mostrados na Tabela 5.1. 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 5.1, 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.

Tabela 5.1: O Algoritmo Híbrido

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

## 5.5

### Resultados de Simulação

Os resultados que serão mostrados a seguir referem-se à codificação dos mesmos objetos e às mesmas condições descritas na Seção 4.2. Os blocos de contorno foram codificados usando a DCT baseada em blocos após haverem 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  $RPR \times bppo$  e os valores de  $RPR$  foram medidos apenas sobre os blocos de contorno.

As Figuras 5.27 a 5.32 apresentam as curvas de  $RPR \times bppo$  para cada objeto. Para o cômputo da taxa  $bppo$ , 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 [80], ou MALV, como proposto em [77]. 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 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.

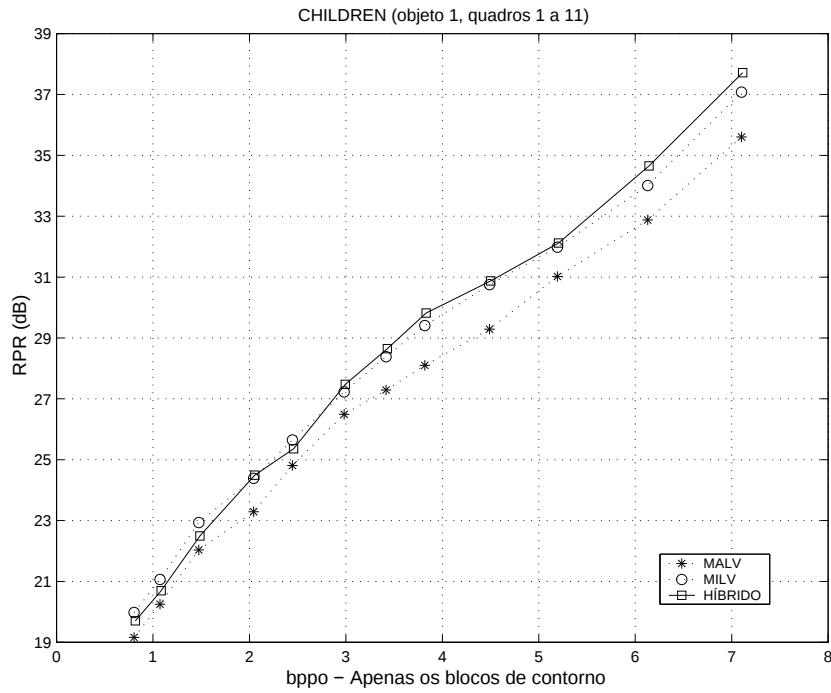


Figura 5.27: Desempenho das diversas estratégias EI-DCT para os 11 quadros do objeto 1 de *CHILDREN*

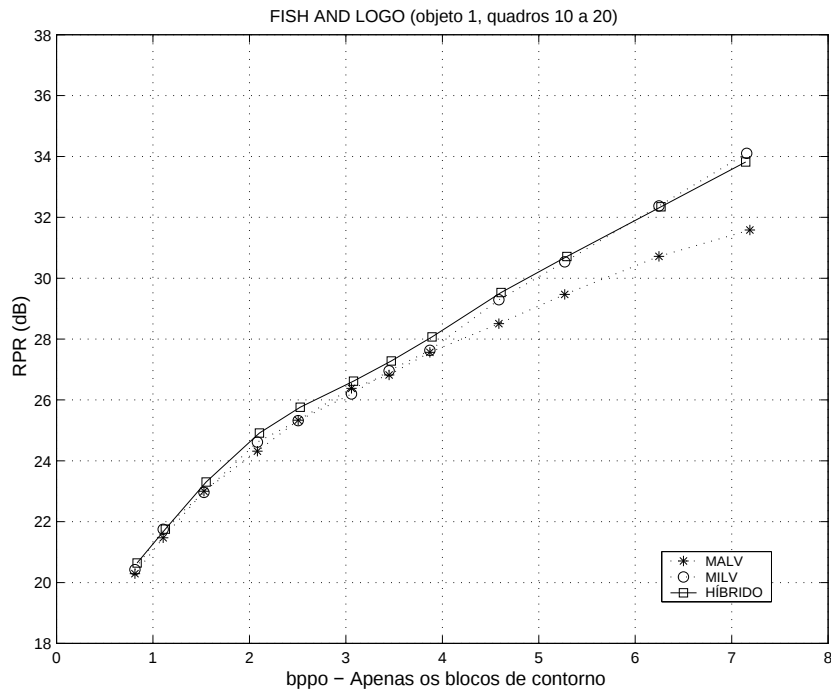


Figura 5.28: Desempenho das diversas estratégias EI-DCT para os 11 quadros do objeto 1 de *FISH AND LOGO*

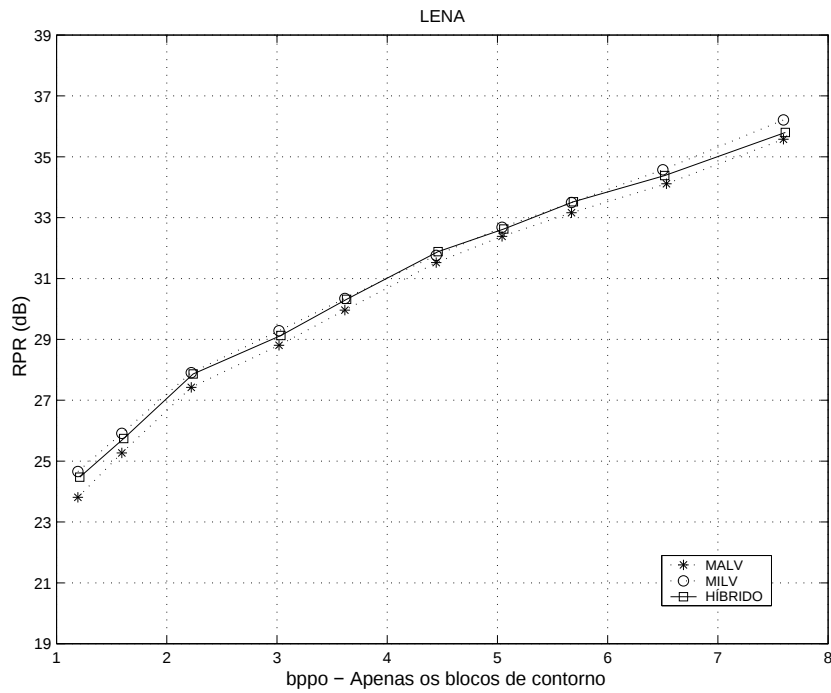


Figura 5.29: Desempenho das diversas estratégias EI-DCT para o objeto de *LENA*

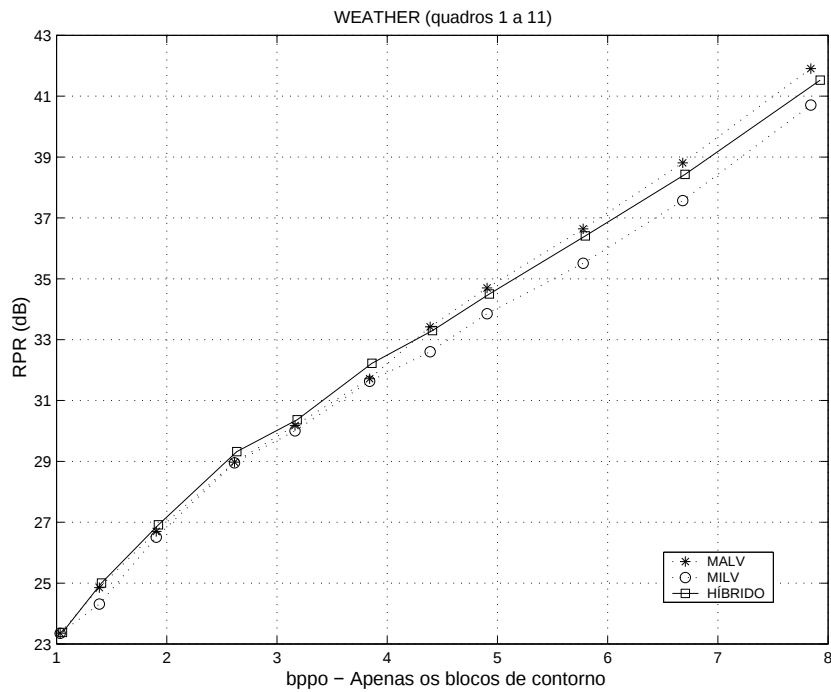


Figura 5.30: Desempenho das diversas estratégias EI-DCT para os 11 quadros do objeto de *WEATHER*

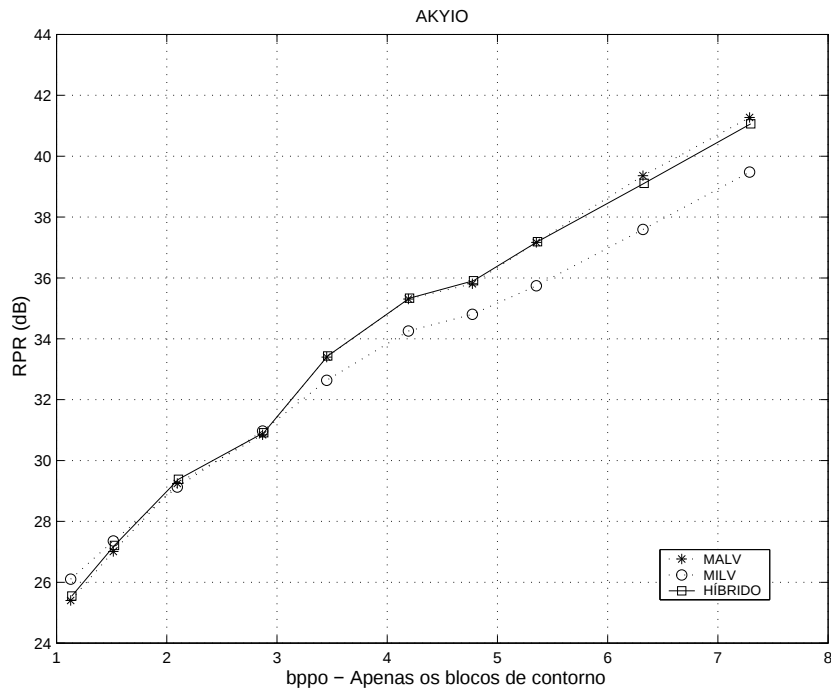


Figura 5.31: Desempenho das diversas estratégias EI-DCT para o objeto de *AKYIO*

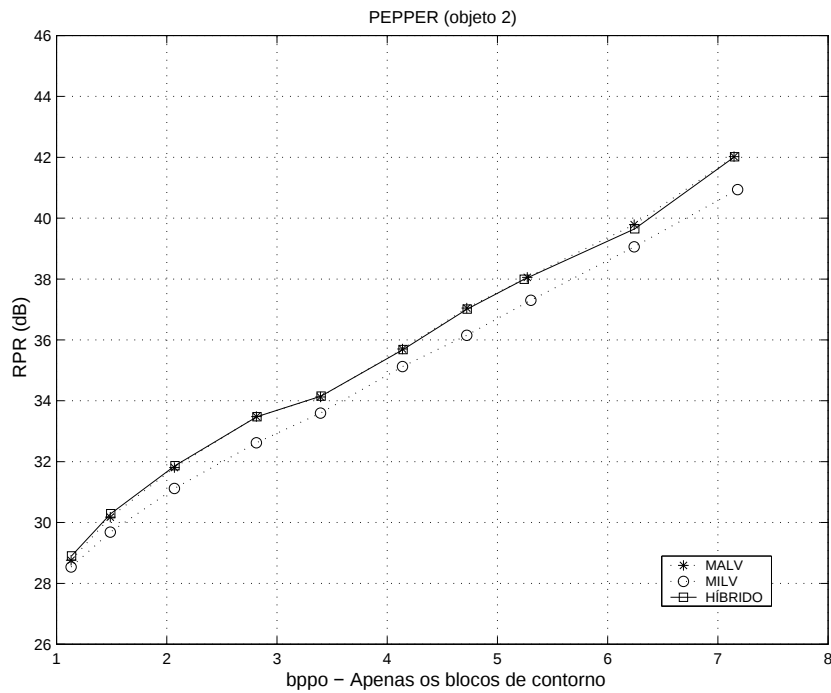


Figura 5.32: Desempenho das diversas estratégias EI-DCT para o objeto 2 de *PEPPER*

## 5.6

### Resultados com Particionamentos Ortogonal e Flexível

Nesta seção, aplicaremos os algoritmos de partição da imagem em blocos [76] que foram apontados, no Capítulo 3, 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 [94]. Ambos têm por objetivo principal a redução do número de blocos de contorno ( $NBC$ ), mas também são eficientes para a redução do número total de blocos ( $NTB$ ), o que aumenta a Área Efetiva dos objetos. 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.

Uma outra vantagem apontada para os esquemas de particionamento [76] é a necessidade da transmissão de um menor número de vetores de movimento para a predição de quadros na codificação INTER-QUADROS (não contemplada nesta tese). Como no algoritmo Flexível está previsto o cálculo de uma referência horizontal diferente para cada camada de blocos (o que corresponde a uma ou mais camadas de blocos  $8 \times 8$  em um quadro de imagem), deve-se garantir que não sejam adotadas referências horizontais distintas para linhas consecutivas de blocos que apresentem blocos compartilhando um mesmo vetor de movimento. Se a compensação de movimento utiliza apenas um vetor por macrobloco, significa que as referências horizontais devem ser iguais para duas linhas de blocos consecutivas. Contudo, como o MPEG-4 - Parte 2 pode empregar quatro vetores de movimento para cada macrobloco, ou seja, um vetor para cada bloco  $8 \times 8$ , a referência horizontal de cada camada de blocos pode ser calculada de forma independente, aumentando as chances de redução do número de blocos. Os resultados obtidos com os dois novos particionamentos aplicados sobre a estratégia híbrida proposta serão comparados nesta seção àqueles obtidos com a partição convencional, aplicada sobre a mesma estratégia.

A Tabela 5.2 apresenta os valores da Área Efetiva para os objetos codificados sob os particionamentos Convencional, Ortogonal Simplificado e Flexível Simplificado. Também são mostrados os valores da Área Efetiva ao se excluirmos, 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.

Tabela 5.2: Áreas Efetivas dos objetos sob diferentes esquemas de particionamento

| OBJETOS              | $AE$<br>(todos os blocos) |              |              | $AE_{BC}$<br>(apenas blocos de contorno) |              |              |
|----------------------|---------------------------|--------------|--------------|------------------------------------------|--------------|--------------|
|                      | $CONV.$                   | $ORT_{simp}$ | $FLX_{simp}$ | $CONV.$                                  | $ORT_{simp}$ | $FLX_{simp}$ |
| <i>CHILDREN</i>      | 63,8                      | 65,6         | 72,0         | 46,7                                     | 48,9         | 57,0         |
| <i>FISH AND LOGO</i> | 66,8                      | 69,4         | 72,8         | 48,0                                     | 49,0         | 52,6         |
| <i>WEATHER</i>       | 85,1                      | 86,2         | 90,3         | 48,4                                     | 39,5         | 50,8         |
| <i>AKYIO</i>         | 93,0                      | 93,3         | 95,0         | 48,5                                     | 45,8         | 54,0         |
| <i>LENA</i>          | 95,5                      | 95,7         | 96,7         | 47,1                                     | 47,6         | 54,0         |
| <i>PEPPER</i>        | 91,6                      | 92,9         | 94,1         | 48,3                                     | 49,8         | 54,2         |

A partir da Tabela 5.2, 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 altera todos os indicadores morfológicos do objeto, e não apenas a Área Efetiva. Com a modificação do indicador  $TNPO$ , pode ocorrer que um determinado grupo de blocos utilize uma estratégia para a determinação da direção preferencial com um particionamento específico, mas que aplique outra estratégia, com um particionamento distinto. Contudo, os limiares de chaveamento entre as estratégias MALV, MACES E MILV no esquema híbrido, apesar de haverem sido calibrados para o particionamento convencional, continuam mantendo a eficiência do algoritmo proposto também para os outros particionamentos. A partir dos resultados mostrados nas Figuras 5.33 a 5.38, 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.

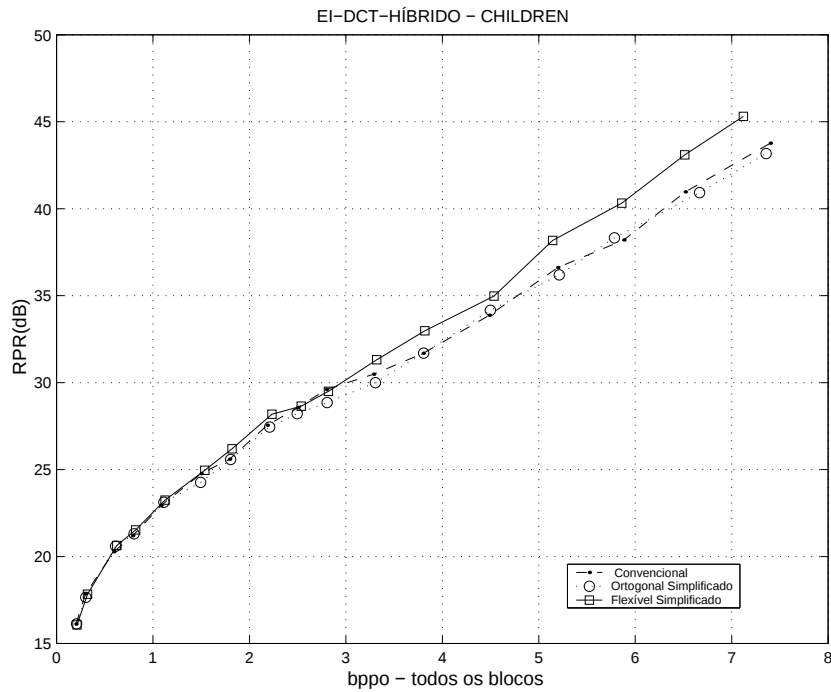


Figura 5.33: Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *CHILDREN* (objeto 1)

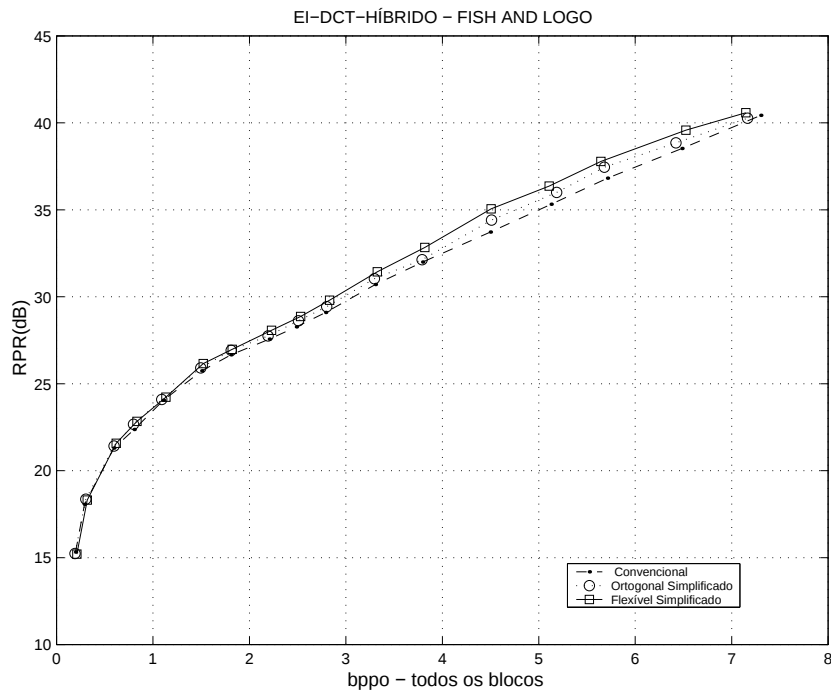


Figura 5.34: Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *FISH AND LOGO* (objeto 1)

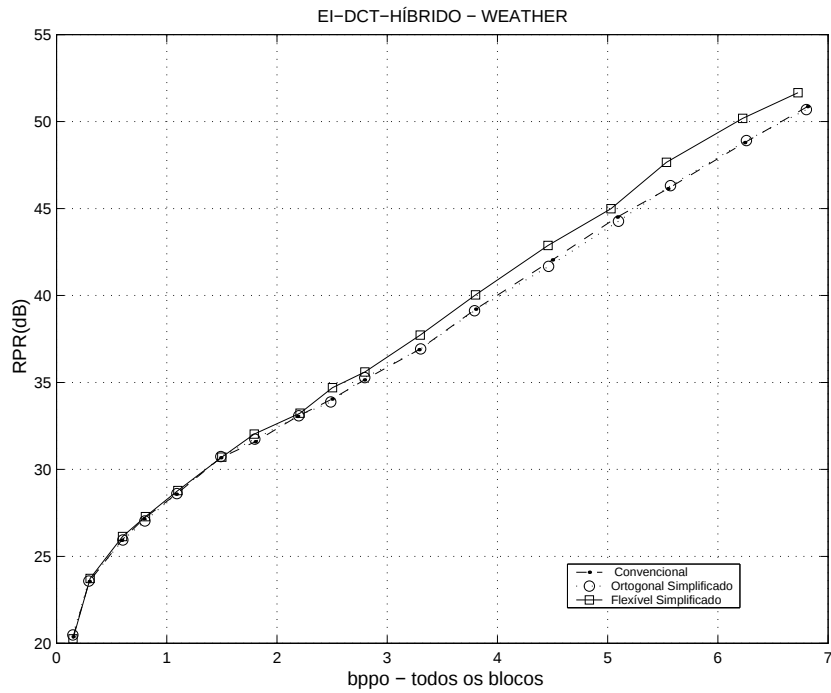


Figura 5.35: Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *WEATHER*)

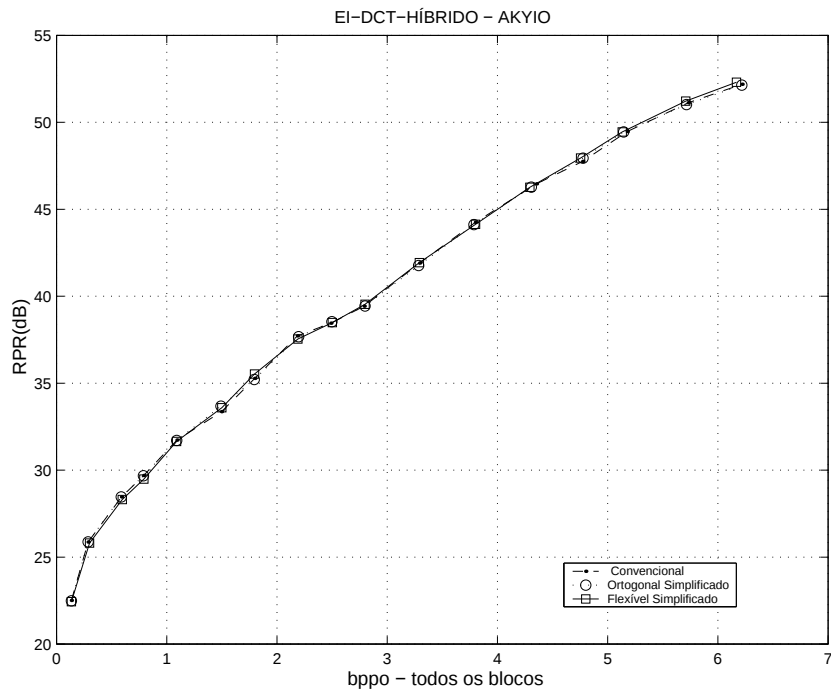


Figura 5.36: Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *AKYIO*

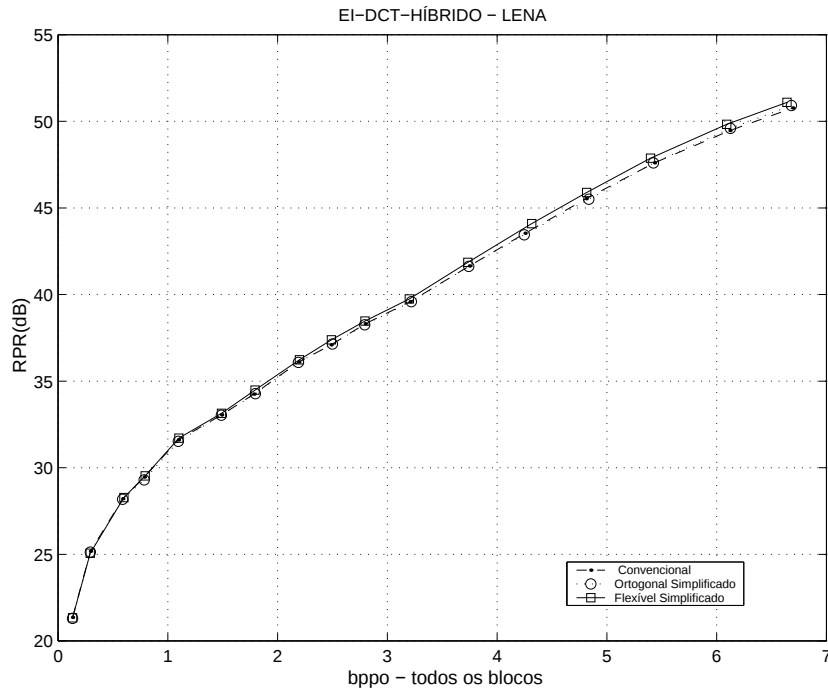


Figura 5.37: Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *LENA*

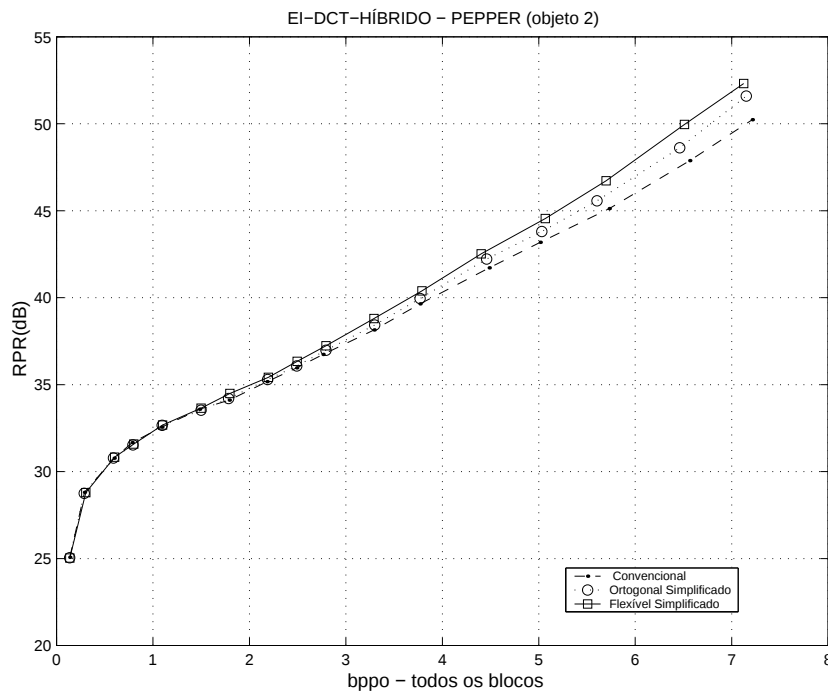


Figura 5.38: Desempenho da estratégia HÍBRIDA associada a diferentes particionamentos - *PEPPER* (objeto 2)

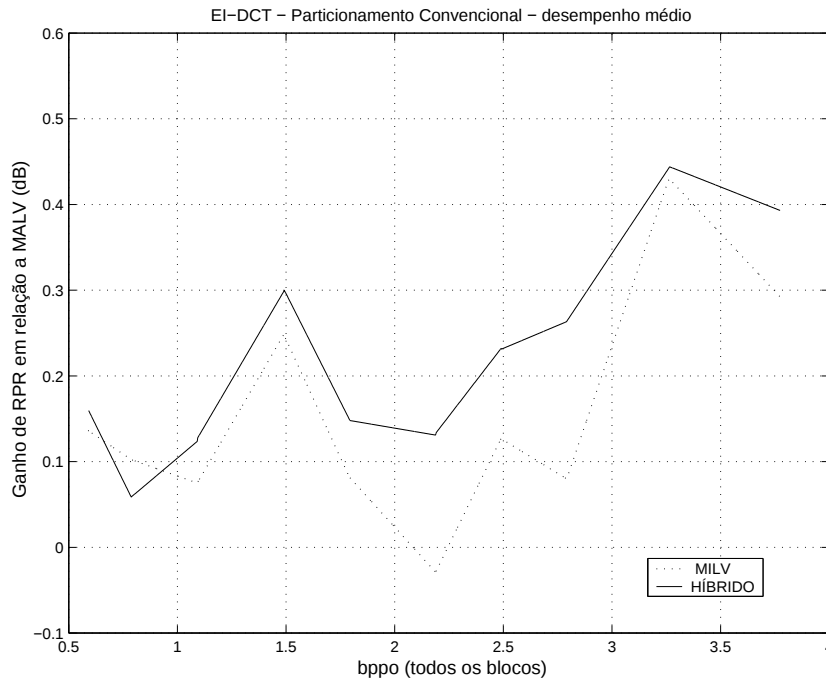


Figura 5.39: Ganhos de desempenho médios sobre a estratégia MALV no Particionamento Convencional

Os gráficos das Figuras 5.39 a 5.41 mostram os ganhos médios de  $RPR$  apresentados pelas estratégias MILV e HÍBRIDA sobre a estratégia MALV nos particionamentos Convencional, Ortogonal Simplificado e Flexível Simplificado. Para a composição dessas curvas, tomou-se o valor médio de codificação 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, onde 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.

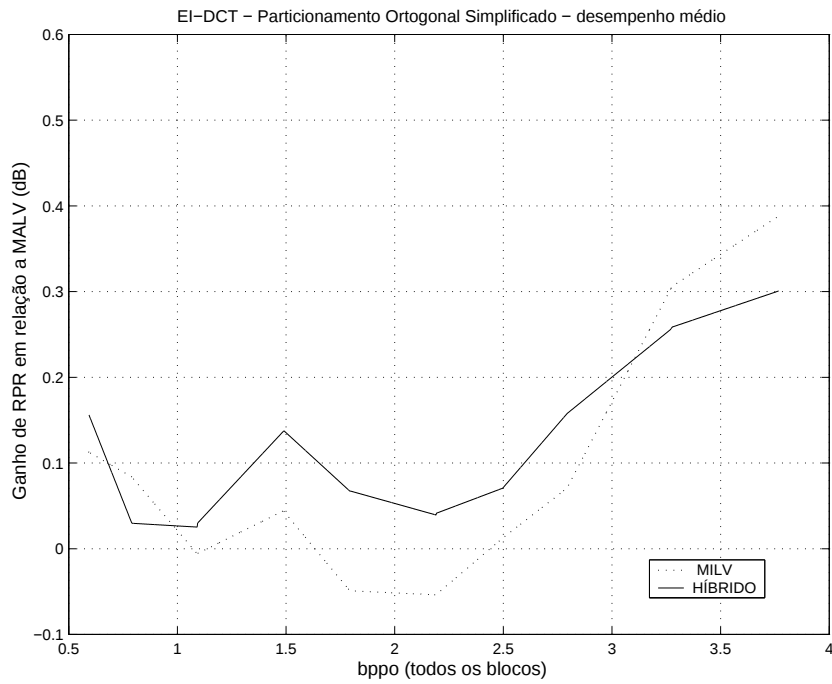


Figura 5.40: Ganhos de desempenho médios sobre a estratégia MALV no Particionamento Ortogonal Simplificado

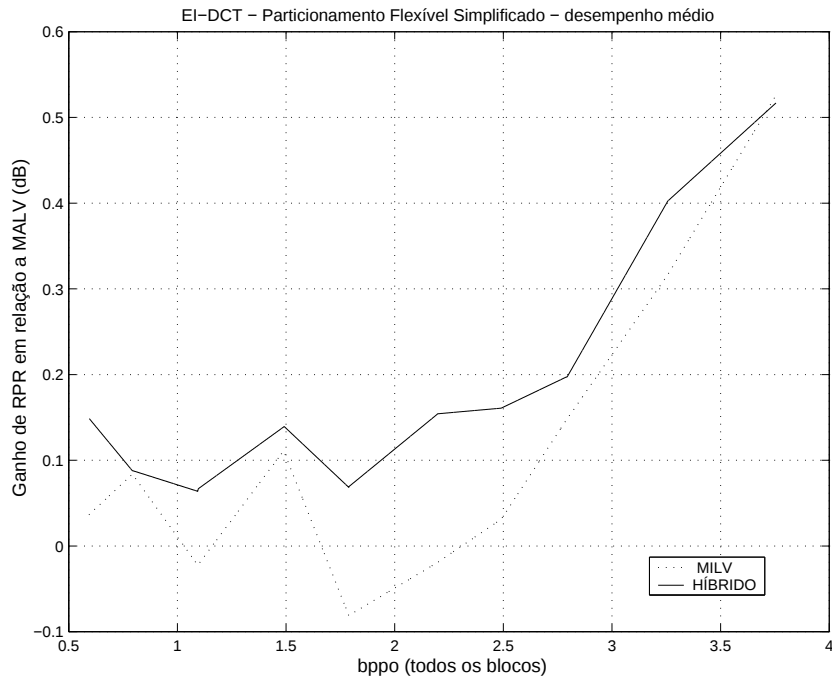


Figura 5.41: Ganhos de desempenho médios sobre a estratégia MALV no Particionamento Flexível Simplificado

## 5.7

### Conclusões

Este capítulo examinou inicialmente 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 [80] o algoritmo MILV foi usado para decidir a prioridade de direção para EI, mas em [77] foi apontado que a eficiência de MALV é melhor que a de MILV. Na realidade, nenhuma das duas afirmações é verdadeira. Neste capítulo 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. 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 (*MAximum Cumulative Energy Sum*), 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 da

Seção 5.3, 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.

Foi proposto um esquema híbrido que 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 capítulo foi mostrado também que o esquema híbrido pode ser usado em conjunto com os esquemas de partição da imagem analisados no Capítulo 3. 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 utilizar os particionamentos Convencional, Ortogonal Simplificado e Flexível Simplificado.