

# COMPRESSÃO DE SEQÜÊNCIAS DE IMAGENS COM ALTA QUALIDADE E BAIXA TAXA DE BITS

MARCUS V. LAMAR, JOSÉ C. M. BERMUDEZ E R. SEARA

Departamento de Engenharia Elétrica, Universidade Federal do Paraná, Curitiba, PR, 80001-970, Brasil,

Laboratório de Instrumentação Eletrônica (LINSE), Departamento de Engenharia Elétrica,

Universidade Federal de Santa Catarina, Florianópolis, SC, 88040-900, Brasil.

**Resumo** Este artigo apresenta uma nova estrutura de codificação para a compressão de seqüências de imagens em sistemas de videofone e videoconferência. A nova estrutura utiliza a segmentação da imagem por decomposição quadtree na etapa de estimação e compensação de movimento. Este procedimento leva a uma redução significativa do esforço computacional de codificação. O novo codificador proporciona a obtenção de uma qualidade objetiva elevada para a operação a baixas taxas de bits. O desempenho da estrutura proposta é comparado, através de simulações por computador, com o desempenho de um codificador baseado na recomendação H.261. Os resultados obtidos demonstram a eficiência do novo esquema de codificação.

**ABSTRACT** This paper presents a new video coding structure for application in videophone and videoconferencing systems. The new structure utilizes image segmentation by quadtree decomposition in the motion estimation and compensation stages. A significant reduction in computational effort is achieved, as compared to conventional structure. The new scheme leads to a high objective image quality for operation at low bit rates. Computer simulations are presented which compare the new structure and a coder based on the H.261 standard for performance. The results obtained demonstrate the effectiveness of the new coding scheme.

Palavra Chave : Compressão de Vídeo, Codificação a baixa taxa, Decomposição Quadtree.

## 1. INTRODUÇÃO

A necessidade da compressão de seqüências de imagens tem crescido bastante nos últimos tempos. Tal fato deve-se a fatores como as proliferações de computadores pessoais e de sistemas multimídia, e a demanda da sociedade moderna por sistemas de comunicações rápidos, eficientes e de baixo custo. Diversos esquemas de codificação de vídeo têm sido propostos procurando atender às mais diferentes aplicações. As aplicações são basicamente classificadas de acordo com a relação entre taxa de bits e qualidade da imagem requeridas no procedimento de codificação. Como exemplos, podemos citar a televisão de alta definição (HDTV) com taxa de bits da ordem de 20 Mbits/s [18]; o padrão MPEG 2, para sistemas de televisão comercial, com taxa de bits em torno de 10 Mbits/s [18]; o padrão MPEG 1, para aplicações em vídeo digital, com taxa de bits da ordem de 1,5 Mbits/s [18]; e o padrão H.261, para codificação em aplicações de videofone e videoconferência, com baixa taxa de bits,  $p \times 64$  kbit/s onde  $p=1,2,\dots,30$  [16-18], dentre outros. Continuando nesta mesma linha, poderíamos ainda destacar o esforço envidado para a concepção de novos padrões de codificação, sendo um deles o MPEG 4, a ser utilizado na compressão de sinais de vídeo a muito baixa taxa de bits (abaixo de 64 kbits/s) [18].

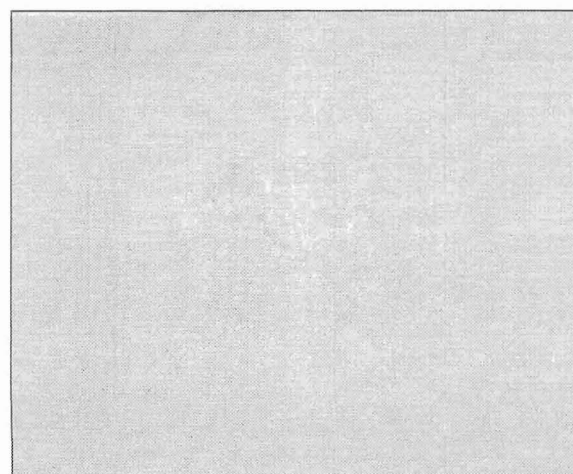
Assim, os esforços de pesquisa têm sido direcionados para a obtenção de codificadores eficientes para a compressão de seqüências de imagens. Tais esquemas de codificação procuram aliar baixa taxa de bits com alta qualidade de imagem.

A obtenção de altas taxas de compressão requer estruturas de codificação que explorem tanto redundâncias espaciais (intra-quadros) como redundâncias temporais existentes entre quadros consecutivos (inter-quadros). O emprego de técnicas de estimação e compensação de movimento na codificação inter-quadros é uma estratégia fundamental para a compressão de sinais de vídeo a baixas taxas de bits. Estas técnicas permitem uma eficiente remoção de redundâncias temporais existentes em uma seqüência de imagens.

Uma das técnicas mais utilizadas para tratar redundâncias inter-quadros é a codificação das *imagens diferença entre quadros com compensação de movimento* (DCM) [4,5]. Essas imagens possuem características bem diversas das imagens dos quadros. O alto índice de correlação existente entre quadros e a possível compensação de movimentos levam à obtenção de DCM com níveis muito baixos de energia. Além disso, a energia das DCM concentra-se nos contornos dos objetos em movimento. A Fig.1a mostra um exemplo de uma imagem obtida pela simples diferença entre dois quadros consecutivos. A Fig.1b mostra a correspondente diferença após a aplicação da compensação de movimento (DCM).



(a)



(b)

Fig. 1. Exemplos de imagens diferença. (a) Diferença entre dois quadros consecutivos. (b) Diferença entre dois quadros consecutivos com compensação de movimento (DCM).

Recentemente, diversos trabalhos de pesquisa têm discutido a utilização de métodos baseados em transformadas (por exemplo, *Discrete Cosine Transform* (DCT) [1]) para a codificação das DCM. No entanto, a dificuldade de obtenção de uma taxa de compactação eficaz dessas imagens, utilizando transformadas, tem orientado as pesquisas para a utilização de estruturas de codificação que atuem diretamente no domínio espacial [1,5,10].

Uma técnica recente que exibe um grande potencial para a representação de imagens em diferentes níveis de resolução é a decomposição quadtree (QT). Esta decomposição foi empregada por Strobach [1,3] com o objetivo de reduzir o volume de dados das DCM a serem quantizadas. Mais recentemente, outros autores têm procurado aprimorar o desempenho da técnica de decomposição QT aplicada à codificação de sinais de vídeo [2,4,5,10,11,13].

Até o presente, a decomposição QT tem sido utilizada apenas na segmentação das DCM, visando à redução dos dados a serem quantizados. Mantém-se, no entanto, a necessidade de determinação dos vetores de deslocamento (VD) do estimador de movimento para o quadro inteiro. A determinação de todos os VD acarreta um esforço computacional significativo.

Este artigo apresenta uma nova estrutura para a codificação de seqüências de imagens. A nova estrutura aplica a técnica de decomposição QT na determinação dos VD do estimador de movimento [19]. Esta estrutura de codificação reduz substancialmente o esforço computacional na etapa de estimação de movimento. Além disso, obtém-se uma redução na taxa de bits para a mesma qualidade de codificação, quando comparada ao esquema de codificação sugerido pela recomendação H.261 [16].

A Seção II apresenta uma síntese da filosofia da decomposição QT, destacando alguns aspectos relevantes para a sua aplicação em codificação de imagens. A Seção III apresenta a nova estrutura de codificação, que emprega a decomposição

QT nas etapas de estimação e compensação de movimento. A Seção IV aborda a forma de quantização utilizada na codificação das folhas da QT. A Seção V apresenta e discute os resultados de simulações. Também é apresentada uma comparação entre a nova estrutura de codificação e um codificador baseado na recomendação H.261. Finalmente, na Seção VI são destacadas as principais conclusões deste trabalho.

## 2. DECOMPOSIÇÃO QT

A decomposição QT é uma simples e bem conhecida técnica para representação de imagens com diferentes níveis de resolução. Inicialmente, essa representação foi usada com sucesso na compressão de imagens binárias. Em seguida, foi aplicada em codificadores de imagens contendo diversos níveis de cinza. Recentemente, a decomposição QT está sendo empregada em algoritmos de compressão de seqüências de imagens [1,3]. A decomposição QT para a codificação de imagens é sedutora pelas seguintes razões:

- i) apresenta uma relativa simplicidade quando comparada a outros métodos (por exemplo, técnicas de compressão baseadas na DCT);
- ii) exibe uma excelente adaptatividade de segmentação;
- iii) o resultado da segmentação é diretamente disponível e pode ser utilizado por uma grande variedade de aplicações em processamento de imagens.

Uma imagem pode ser dividida (segmentada) em regiões de diferentes tamanhos com uma quantidade variável de detalhes e informações. Essa segmentação é útil para uma eficiente codificação dos dados da imagem. A decomposição QT é uma técnica poderosa que segmenta a imagem em regiões bidimensionais homogêneas (com respeito à propriedade de interesse).

A decomposição QT dá origem a uma árvore. Cada nó dessa árvore origina quatro blocos-filhos e está associado a uma única região da imagem. O nó-raiz da árvore é associado à imagem completa. A decomposição QT pode ser efetuada através de um procedimento denominado de cima-para-baixo ou através do procedimento inverso, de baixo-para-cima. A abordagem de cima-para-baixo consiste de sucessivas subdivisões da imagem original por um fator quatro. Se um bloco é homogêneo (com relação a algum critério dado), ele não é subdividido, caso contrário ele será subdividido em quatro sub-blocos. A seqüência de subdivisões de blocos não-homogêneos continua até o menor bloco atingir um tamanho mínimo conveniente. Estes blocos de tamanho mínimo são denominados folhas da decomposição QT. Na técnica de baixo-para-cima, parte-se da imagem completa dividida em blocos de tamanho mínimo (as folhas). Então testa-se a homogeneidade de quatro blocos adjacentes. Se eles forem homogêneos, são reunidos formando um bloco maior. A Fig. 2 ilustra estas duas diferentes formas de decomposição QT. A Fig.3 mostra a estrutura da árvore resultante da decomposição QT ilustrada pela Fig. 2.

Um ponto fundamental no processo de decomposição QT é o critério usado para definir a homogeneidade de um bloco de pixels. Esse é o critério que determina a atividade de um bloco da imagem. Existem diversas propostas para a medida da atividade [6]. Uma das mais usuais é a variância das intensidades dos pixels dentro do bloco. Se o valor da variância estiver abaixo de um dado limiar, o bloco é considerado homogêneo. Uma medida de atividade mais simples usa a média das intensidades dos pixels. O bloco é considerado homogêneo se todas as intensidades de seus pixels são suficientemente próximas (dada uma tolerância máxima) ao valor médio da intensidade do bloco inteiro.

No procedimento de baixo-para-cima pode-se utilizar as médias e as variâncias já calculadas nas etapas anteriores para a determinação de seus valores atuais. Por isso, este procedimento é computacionalmente muito mais eficiente do que o da decomposição de cima-para-baixo [1].

Estas medidas de atividade são, em geral, muito eficientes. No entanto, para imagens obtidas pela diferença de quadros consecutivos em uma seqüência de imagens, estas medidas levam a uma grande quantidade de cálculos desnecessários. Estas imagens diferença são compostas predominantemente por pixels com intensidades muito baixas ou muito altas (Fig.1a). Intensidades muito baixas ocorrem em regiões onde não houve movimento. Estes níveis de intensidades são resultantes, principalmente, dos ruídos de aquisição e quantização. Regiões onde ocorreu movimento tendem a conter pixels com intensidades muito altas. São essas as regiões que devem ser codificadas e as denominaremos daqui em diante por *regiões ativas*.

Regiões ativas têm pixels com altos níveis de intensidade, quando comparados com os níveis de intensidade dos pixels das regiões não-ativas. Assim, pode-se efetivamente empregar um critério de homogeneidade muito simples: a comparação com um nível limiar pré-definido. Uma comparação direta desse limiar com a intensidade do pixel pode decidir se o pixel é ativo ou não. Regiões ativas são aquelas contendo pixels ativos. Regiões homogêneas são regiões não-ativas. É

importante observar que este critério requer apenas uma simples comparação para cada pixel do bloco. Além do mais, deve-se considerar que um bloco possuindo um único pixel com intensidade acima do nível limiar já deve ser subdividido. Portanto, a ocorrência de um pixel ativo elimina a necessidade de teste dos demais pixels do bloco. Desta forma, obtém-se uma complexidade computacional reduzida nesta etapa.

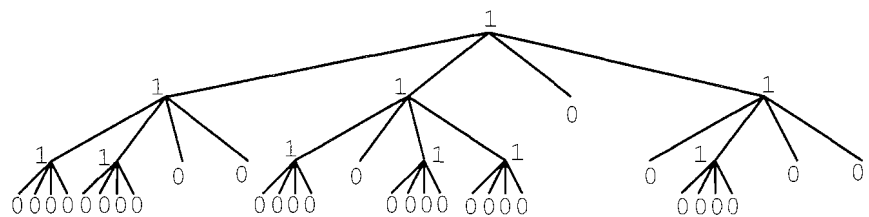
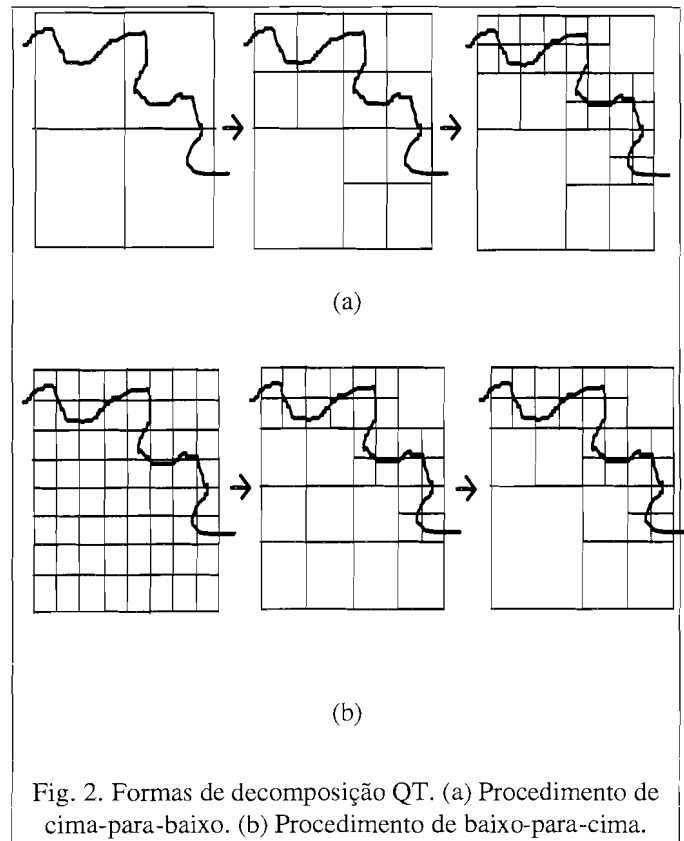
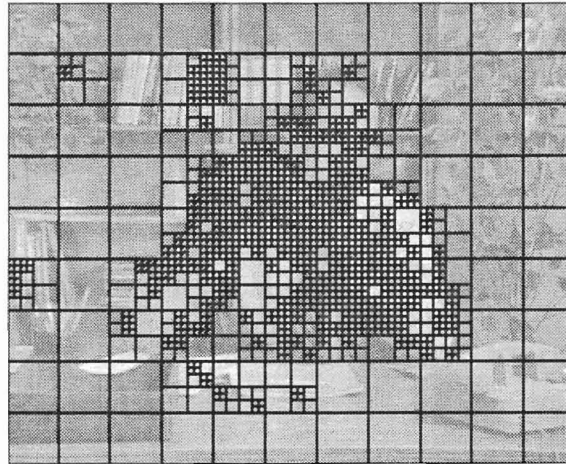


Fig. 3. Estrutura da árvore resultante da decomposição ilustrada pela Fig. 2.

A Fig. 4a mostra um exemplo de segmentação por decomposição QT da imagem diferença entre o quadro atual e uma estimativa do quadro anterior. Inicialmente, dividiu-se a imagem em blocos de 32x32 pixels. Essa divisão permite adequar as dimensões da imagem às exigências da decomposição QT regular. O nó raiz terá dimensão de potência de 2, possibilitando a aplicação recursiva do algoritmo de decomposição. A dimensão das folhas foi definida em 4x4 pixels. A Fig. 4b apresenta a estrutura de bits correspondente a esta decomposição. Nessa representação convencionou-se que o bit de valor 1 está indicando que o bloco deve ser subdividido. O bit de valor 0 indica que o bloco não deve ser mais subdividido. Não estão representados aqui os bits de valor 0 correspondentes às folhas, pois esses bits não carregam qualquer informação útil para o processo de decomposição. A informação do tamanho da estrutura de bits da QT é incluída adicionando-se 11 bits no início dessa estrutura. Esta informação é necessária para a correta descodificação da QT na recepção.

Tirando proveito dos conceitos e propriedades psicofísicas do mecanismo da visão humana [15], poderíamos aumentar o nível limiar de decomposição até um valor determinado pela sensibilidade do aparelho visual humano. Isso reduziria ainda mais a complexidade da decomposição QT. No entanto, a determinação de um nível limiar ótimo baseado nas propriedades psicovisuais humanas está além do escopo deste artigo.



(a)

675

```

000000000000101111000000011111000000111110000001111100101111111000011111
110000011100000000000000010000101101001010110010000011111111100100101111
11111111100110111111111111111101110101010010111011010111111111110111000
111001011111111111011111010000110001001001111111010101000010100001000100
00100110001001100101100000101101100111110000001011111111111110101111110
101111111101011011111111001000011111001111101111111111111111101111111110
111111111101110111110101111110011010000111101000110101001110110100010111
11101000110100011111011111111111111111101101111111010111001010010101010
1111100100001001101100001011001101111111000100011111010110001011010101000
101001100000011000

```

(b)

Fig. 4 Exemplo de segmentação por QT. (a) Decomposição QT. (b) Estrutura de bits da QT

### 3. DECOMPOSIÇÃO QT APLICADA À ESTIMAÇÃO E COMPENSAÇÃO DE MOVIMENTO

A Fig. 5 apresenta a estrutura do esquema proposto de codificação, utilizando segmentação por decomposição QT para a etapa de *estimação de movimento* (EM). Primeiramente, a *diferença entre quadros* (DF) é determinada subtraindo do *quadro atual*  $F(n)$  o *quadro anterior codificado*  $\hat{F}(n-1)$ . A DF é então segmentada por decomposição QT utilizando um nível limiar adequado para a determinação das regiões ativas. O bloco EM determina os VD apenas para as regiões ativas da DF. Como as regiões ativas representam uma pequena porcentagem da DF, o esquema de codificação proposto apresenta uma grande eficiência computacional na etapa de estimação de movimento. O bloco CM efetua a compensação de movimento a partir dos VD da estrutura da QT e do quadro  $\hat{F}(n-1)$ . O quadro  $\hat{F}(n-1)$  é modificado de modo a gerar uma *predição*  $C[\hat{F}(n-1)]$  do quadro atual. A imagem diferença entre as regiões de  $F(n)$  e  $C[\hat{F}(n-1)]$  selecionadas pela QT é denominada *erro de predição* (EP). O EP é quantizado através do bloco Q. O receptor primeiramente recebe o *bit stream* da QT de modo a reconstitui-la. A seguir são enviados os VD correspondentes às regiões selecionadas e, finalmente, os dados relativos às regiões quantizadas. Dada a estrutura de decomposição QT, define-se que todos os pixels fora das regiões selecionadas (blocos ativos) da DF possuem valor nulo, isto é, não tiveram movimento. Os dados relativos à quantização das regiões selecionadas são reconstruídos pelo bloco  $Q^{-1}$  ( $Q^{-1}$  representa a operação de quantização inversa) e somados ao quadro  $C[\hat{F}(n-1)]$ . Essa operação dá origem ao quadro atual estimado  $\hat{F}(n)$ , o qual será utilizado na codificação do quadro seguinte.

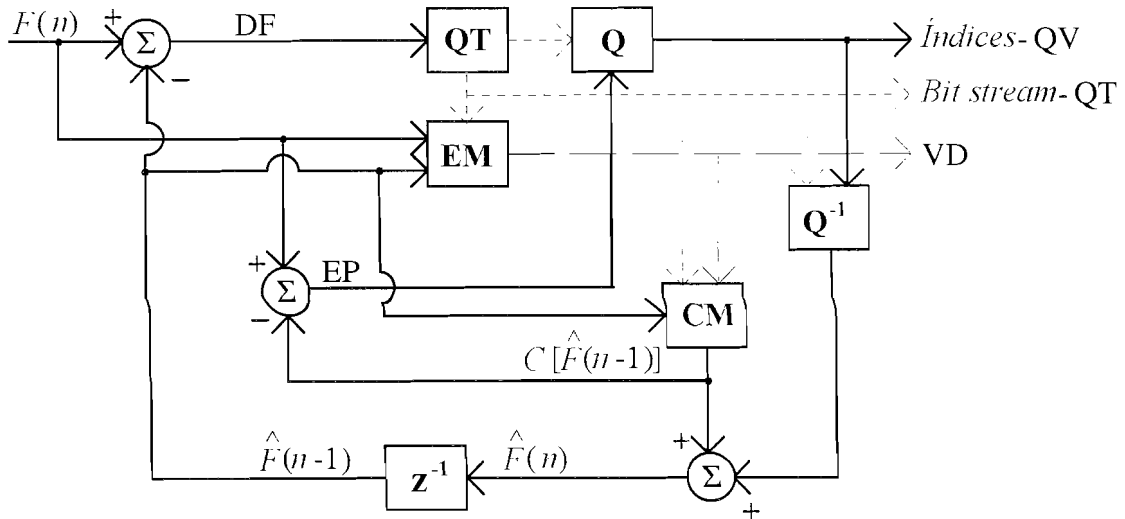


Fig. 5. Estrutura do codificador proposto.

O algoritmo de estimação de movimento é fundamental para uma compensação de movimento eficiente. Neste trabalho utilizamos o já conhecido *Block Matching Algorithm* (BMA) [8-7] devido à sua eficiência e simplicidade de implementação. Outros algoritmos do tipo *block matching* podem também ser empregados [14].

O BMA baseia-se na suposição de que o movimento de um objeto rígido em uma seqüência de quadros é puramente translacional (paralelo ao plano da câmera). Inicialmente, subdivide-se o quadro em blocos não sobrepostos (tipicamente de tamanho 16x16). Para cada bloco selecionado do quadro atual, procura-se onde o referido bloco está localizado no quadro anterior. Esta busca é realizada em uma região definida em torno da posição do bloco selecionado (janela de busca). A melhor localização é aquela que otimiza uma dada função casamento. As posições relativas de blocos casados dão origem às coordenadas dos VD para aqueles blocos. A precisão da estimativa do VD depende basicamente do critério empregado para o casamento de blocos [9] e do tamanho da janela de busca utilizada. Os critérios mais difundidos de casamento de blocos são: *mean square error* (MSE), *pixel difference classification* (PDC), *cross-correlation*, *mean absolute difference* (MAD), dentre outros. Neste trabalho, utilizamos o critério MAD devido à sua reduzida complexidade computacional. O MAD não requer operações de multiplicação e tem um desempenho equivalente ao critério MSE. O MAD entre dois blocos de tamanho  $L \times M$ , referenciados pelas coordenadas  $(x,y)$  e  $(x-i,y-j)$  de seus cantos superiores esquerdos, é definido por:

$$MAD_{x,y}(i,j) = \frac{1}{LM} \sum_{l=0}^{L-1} \sum_{m=0}^{M-1} |I_n(x+l,y+m) - I_{n-1}(x+i+l,y+j+m)| \quad (1)$$

onde  $n$  e  $n-1$  representam os índices dos quadros atual e anterior, respectivamente.  $I_n(l,m)$  caracteriza a intensidade do pixel de coordenada  $(l,m)$ . O critério MAD requer apenas  $2LM$  operações de adição para avaliar o casamento entre dois blocos.

A fim de reduzir ainda mais a complexidade computacional poder-se-ia utilizar alguma técnica de pesquisa rápida na janela de busca [7,8,14]. Tais técnicas não foram utilizadas neste trabalho para não reduzir a qualidade das estimativas dos VD.

Para a estimação de movimento utilizando o algoritmo BMA no contexto do codificador proposto (Fig. 5), deve-se computar a DF e efetuar a decomposição QT sobre esta imagem. As folhas da estrutura da QT correspondem às regiões com pixels com intensidades superiores ao do limiar especificado. A estimação de movimento é realizada para blocos de quatro folhas adjacentes. Essa estratégia utiliza a alta correlação existente entre VD de blocos próximos para diminuir substancialmente a quantidade de vetores necessários. Portanto, se as folhas da QT são de 4x4 pixels, realizaremos a estimação de movimento usando blocos de tamanho 8x8.

Para a compensação de movimento, move-se cada bloco selecionado de sua posição dentro do quadro anterior para a nova posição, de acordo com o respectivo VD. A diferença entre o quadro atual e o quadro com compensação de movimento origina a imagem EP. Considerando que a estrutura da QT define todas as regiões ativas da imagem, é necessário quantizar apenas a imagem correspondente às folhas da QT.

## 4. QUANTIZAÇÃO DAS FOLHAS DA QT

A fim de reduzirmos ao máximo a correlação existente entre pixels das folhas da QT e, desta forma, aproximarmos-nos ao máximo do limite de compressão dado pela teoria de distorção pela taxa, utilizamos quantização vetorial (QV) para codificar o EP.

A QV é definida como o mapeamento de um vetor  $X$  qualquer de dimensão  $K$  em um espaço amostral  $\mathcal{Y}$ , também de dimensão  $K$ , chamado dicionário de códigos (*codebook*). Esse mapeamento é realizado buscando nesse dicionário a melhor representação para o vetor  $X$  segundo a minimização de uma determinada função-distorção. O algoritmo de quantização procura o vetor do dicionário de códigos que minimiza essa função-distorção e guarda (para transmitir) apenas o seu endereço. Uma conveniente escolha para a função-distorção, no caso de imagens, é o erro médio quadrático, definido por:

$$d(X, X_q) = \frac{1}{K} \sum_{k=1}^K [X(k) - X_q(k)]^2 \quad (2)$$

onde  $X$  é o vetor original,  $X_q$  o vetor quantizado e  $K$  a dimensão do espaço.

Para tornar o codificador competitivo a nível de taxa de transmissão, utilizamos um número de bits diferente para definir cada palavra do dicionário de códigos. Deste modo, *codewords* mais frequentes recebem um código com menor número de bits. Esta alocação de bits seria "ótima" se tivéssemos uma estimativa precisa da distribuição de probabilidades das *codewords* das seqüências a serem codificadas. Como isso não é conhecido a priori, optamos, então, por utilizar a própria seqüência de treinamento (que gerou o dicionário de códigos) para determinar uma estimativa das probabilidades de ocorrência destes vetores.

## 5. RESULTADOS DE SIMULAÇÕES

Foram realizados, através de simulações, diversos testes de avaliação da estrutura de codificação proposta. Nesses testes foram utilizadas duas seqüências de imagens bem conhecidas, que são: *Salesman*, com quadros de 352x288 pixels (formato CIF - *Common Intermediate Format*) e *Trevor White* com 256x256. Todas as seqüências eram originalmente amostradas com 30 quadros/s e 8 bits/pixel. Para obter taxas inferiores a 30 quadros/s sub-amostramos convenientemente estas seqüências de imagens.

O dicionário de códigos utilizado na QV foi projetado utilizando o algoritmo LBG [12]. A seqüência de treinamento foi composta por vetores selecionados através da estrutura da QT sobre as DF da seqüência *Trevor White* para uma taxa de amostragem de 7,5 quadros/s. O nível de limiar para a decomposição QT foi arbitrado em 20 (na faixa de 0 a 255). O dicionário de códigos resultante tem 256 palavras-código com dimensão 16. A dimensão 16 é devido ao tamanho de 4x4 pixels usado para as folhas da QT, possibilitando a obtenção de uma taxa de 0,5 bits/pixel. A estratégia de alocação dos códigos dos índices do dicionário foi baseada no *Código de Huffman* [15]. A distribuição de probabilidades necessária à criação do referido código foi obtida a partir da codificação da própria seqüência de treinamento que gerou o dicionário. A Fig. 6 mostra o histograma das palavras-código.

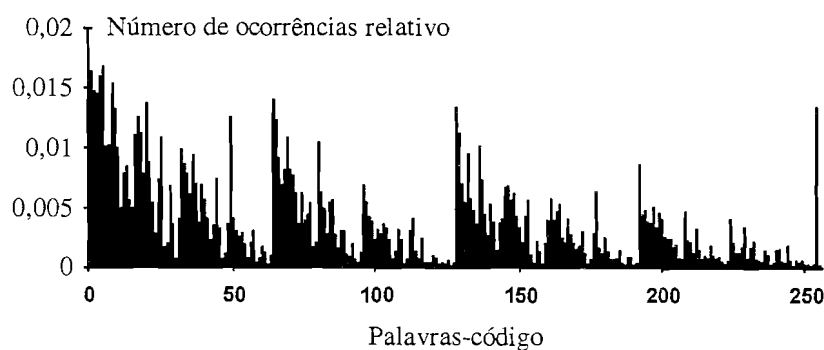


Fig. 6. Histograma das palavras-código.

Com esta alocação de códigos foi possível reduzir a taxa de bits teórica relativa à quantização de  $8/16 = 0,5$  bits/pixel para aproximadamente  $7,27/16 = 0,45$  bits/pixel.

A alocação dos códigos para os vetores de movimento segue a recomendação H.261 [16]. A Tabela I apresenta os códigos que definem cada deslocamento, tanto no sentido vertical quanto no sentido horizontal. Assim, por exemplo, o vetor deslocamento (-1,0) necessitará de (3+1) bits para a sua representação.

TABELA I

Alocações de bits dos códigos de deslocamento.

<i>Deslocamento</i>	<i>Código</i>	<i>Deslocamento</i>	<i>Código</i>
-16	0b00000011001	0	0b1
-15	0b00000011011	1	0b010
-14	0b00000011101	2	0b0010
-13	0b00000011111	3	0b00010
-12	0b00000100001	4	0b0000110
-11	0b00000100011	5	0b00001010
-10	0b0000010011	6	0b00001000
-9	0b0000010101	7	0b00000110
-8	0b0000010111	8	0b0000010110
-7	0b00000111	9	0b0000010100
-6	0b00001001	10	0b0000010010
-5	0b00001011	11	0b00000100010
-4	0b0000111	12	0b00000100000
-3	0b00011	13	0b00000011110
-2	0b0011	14	0b00000011100
-1	0b011	15	0b00000011010

A medida de desempenho adotada neste trabalho é a relação sinal-ruído de pico (*PSNR*), definida por:

$$PSNR = 10 \log \left[ \frac{255^2}{EMQ} \right] \quad \text{dB} \quad (3)$$

onde *EMQ* é o erro médio quadrático, definido por:

$$EMQ = \frac{1}{GH} \sum_{g=0}^{G-1} \sum_{h=0}^{H-1} [I(g,h) - I_c(g,h)]^2 \quad (4)$$

onde *I* e *I<sub>c</sub>* representam, respectivamente, as intensidades dos pixels da imagem original e da imagem codificada, ambas com dimensão de *G*×*H* pixels.

Deve-se ainda destacar que foi utilizada uma taxa de 0,5 bits/pixel para a transmissão do primeiro quadro da seqüência. Isto se faz necessário, visto que a qualidade do primeiro quadro é um fator fundamental para a qualidade da codificação dos quadros subsequentes, influenciando consideravelmente no resultado obtido da *PSNR* média da seqüência. A



codificação do primeiro quadro foi também realizada através de QV, utilizando-se um dicionário de códigos de tamanho 256 com vetores de dimensão 16, projetado a partir das imagens *Lenna*, *Airport* e *Baboom*. Essas condições visam tornar as simulações da estrutura do quantizador desenvolvido tão próximas da realidade quanto possível.

As Tabelas II, III e IV apresentam os valores médios de *PSNR*, taxas de bits e capacidades de canal necessárias aos diferentes valores de níveis de limiar na decomposição QT para taxas de amostragem de 15, 10 e 7,5 quadros/s.

TABELA II

Seqüência *Salesman* - 15 quadros/s.

	<i>limiar</i> = 10	<i>limiar</i> = 15	<i>limiar</i> = 20	<i>limiar</i> = 25
<i>Taxa média</i> [bits/pixel]	0,220235	0,141106	0,106283	0,085646
<i>Canal</i> [kbits/s]	327,05	209,54	157,83	127,18
<i>PSNR média</i> [dB]	35,24	34,08	32,76	31,63

TABELA III

Seqüência *Salesman* - 10 quadros/s.

	<i>limiar</i> = 10	<i>limiar</i> = 15	<i>limiar</i> = 20	<i>limiar</i> = 25
<i>Taxa média</i> [bits/pixel]	0,292766	0,180809	0,133181	0,1055921
<i>Canal</i> [kbits/s]	289,84	179,00	131,85	104,86
<i>PSNR média</i> [dB]	34,64	33,72	32,65	31,55

TABELA IV

Seqüência *Salesman* - 7,5 quadros/s.

	<i>limiar</i> = 10	<i>limiar</i> = 15	<i>limiar</i> = 20	<i>limiar</i> = 25
<i>Taxa média</i> [bits/pixel]	0,271788	0,187606	0,142426	0,116573
<i>Canal</i> [kbits/s]	201,80	139,29	105,75	86,55
<i>PSNR média</i> [dB]	34,47	33,55	32,38	31,38

A Fig. 7 apresenta as curvas de EMQ como função da taxa média de bits, para o codificador proposto e para o codificador baseado na recomendação H.261. Estas características foram obtidas para a seqüência de imagens *Salesman* (formato CIF) sub-amostrada à taxa de 7,5 quadros/s.

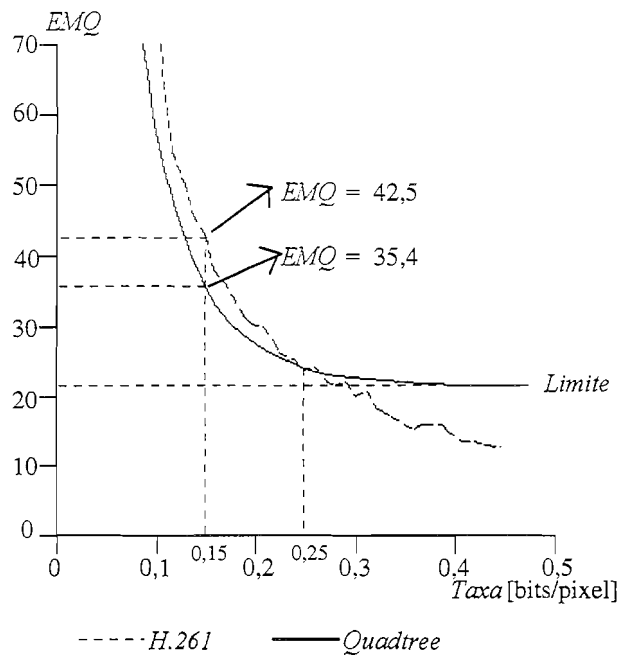


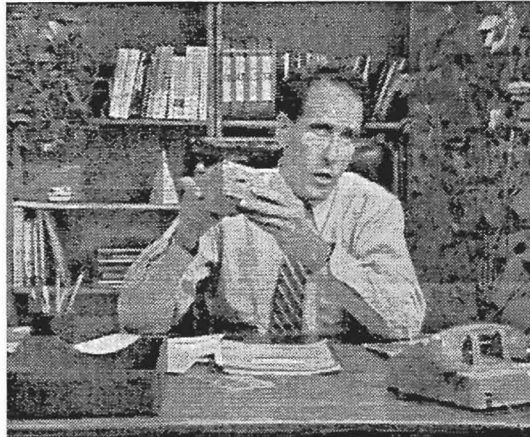
Fig. 7. Distorção versus taxa média de bits para seqüência *Salesman* a 7,5 quadros/s.

Para ambos os codificadores, o tamanho da janela de busca dos VD é  $\pm 7$  pixels, ou seja,  $(-7,-7) \leq (i,j) \leq (7,7)$  na equação (1). O nível limiar para a decomposição QT é variado de acordo com a relação taxa-qualidade desejada, mantendo-se o quantizador vetorial fixo. Da Fig. 7 e da expressão (3) constatamos que o codificador proposto melhora em pelo menos 1 dB a qualidade (PSNR), quando comparado ao codificador baseado na recomendação H.261 para taxas de bits menores ou iguais a 0,15 bits/pixel. A variação do nível limiar na decomposição QT nos permite escolher o ponto de operação sobre a curva de distorção sem precisar alterar a estrutura do quantizador. Para taxas maiores do que 0,25 bits/pixel, a qualidade do codificador baseado na recomendação H.261 é melhor. Este fato é decorrente da alocação de um número maior de bits para a quantização dos coeficientes da DCT no codificador baseado na recomendação H.261. O codificador proposto apresenta um limite mínimo de distorção, abaixo do qual uma melhoria de qualidade requer a modificação da etapa de quantização. Torna-se necessária a utilização de um quantizador vetorial com dicionário de códigos maior.

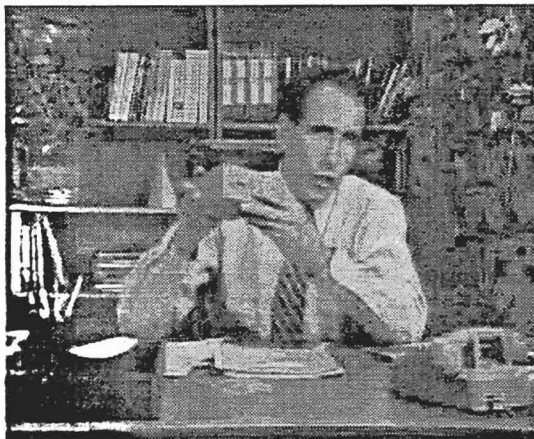
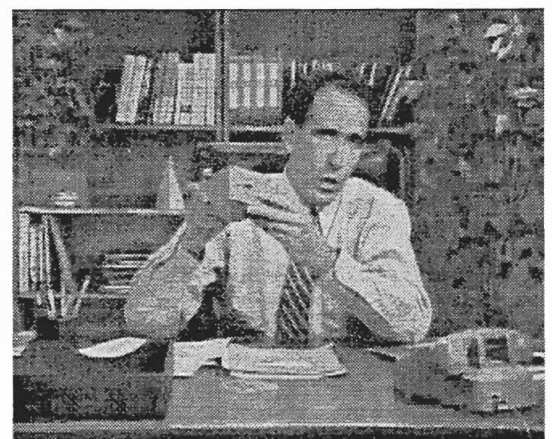
A Fig. 8 apresenta o quadro número 144 da seqüência de imagens *Salesman* para fins de comparação subjetiva. Na Fig. 8a, temos o quadro original, na Fig. 8b, o quadro reconstruído pelo codificador baseado na recomendação H.261 e na Fig. 8c, o mesmo quadro reconstruído pela nova estrutura de codificação. A taxa de bits das imagens codificadas é de 0,11 bits/pixel e a taxa de amostragem da seqüência é de 7,5 quadros/s. Uma análise subjetiva mais justa requer a avaliação de toda a seqüência dinâmica de quadros. Só assim podem ser levadas em consideração todas as características inerentes ao sistema visual humano na observação de imagens em movimento.

## 6. CONCLUSÕES

Este trabalho apresentou um esquema de estimação e compensação de movimento para codificação de vídeo a taxa de bits variável. O novo esquema baseia-se em segmentação pela decomposição QT de imagens diferença entre quadros consecutivos. As regiões ativas do erro de predição são quantizadas vetorialmente. O codificador proposto proporciona a obtenção de uma qualidade subjetiva elevada para operação a baixas taxas de bits, sendo adequado para operações com canais de baixa capacidade de transmissão. A nova estrutura de codificação leva a uma redução considerável de esforço computacional na etapa de estimação de movimento. O desempenho do novo codificador foi comparado com o desempenho de um codificador baseado na recomendação H.261. Obteve-se uma melhoria superior a 1 dB na PSNR para taxas de bits menores ou iguais a 0,15 bits/pixel. A quantização dos resíduos das folhas da QT pode ser ainda melhorada com a utilização de quantizadores vetoriais adaptáveis. Quantizadores adaptáveis levariam em consideração a não-estacionariedade do processo e poderiam ajustar o nível de limiar na decomposição QT para otimizar o dicionário de códigos.



(a) Quadro original

(b) Quadro codificado com codificador da recomendação H.261 -  $PSNR = 30,4$  dB(c) Quadro codificado com a nova estrutura de codificação -  $PSNR = 31,3$  dBFig. 8. Quadro número 144 da seqüência *Salesman* a 7,5 quadros/s e taxa de 0,11 bits/pixel.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Strobach, "Tree-structured scene adaptive coder", *IEEE Trans. on Communications*, vol. 38, pp. 477-486, Apr. 1990.
- [2] M. Nasrabadi, "Use of vector quantizers in image coding", *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Processing*, 1984, pp.10.9.1-10.9.4.
- [3] Strobach, "Quadtree structured recursive plane decomposition coding of images", *IEEE Trans. on Signal Processing*, vol. 39, pp. 1380-1397, June 1991.
- [4] Liu and M. Hayes, "Segmentation-based coding of motion difference and motion field images for low bit-rate video compression", *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Processing*, 1992, pp. III.525-III.528.
- [5] Liu and M. Hayes, "Video compression using quadtree segmentation and component quantization", *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Processing*, 1993, pp. V.429-V.432.
- [6] F. MacLean and M. E. Jernigan, "Indicator functions for adaptive image processing", *J. Opt. Soc. Am.*, vol. 8, Jan. 1991.

- [7] G. Musmann, P. Pirisch and H. J. Grallert, "Advances in picture coding", *Proceedings of the IEEE*, vol. 73, pp.523-548, Apr. 1985.
- [8] R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding", *IEEE Trans. on Communications*, vol. COM-29, pp. 1799-1808, Dec. 1981.
- [9] Gharavi and M. Mills, "Blockmatching motion estimation algorithms - new results", *IEEE Trans. on Circuits and Systems*, vol. 37, pp.649-651, May 1990.
- [10] Lu and W. A. Pearlman, "Multi-rate video coding using pruned tree-structured vector quantization", *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Processing*, 1993, pp.V.253-V.256.
- [11] S. Wu and J. Barba, "Coding of quadtree structures using run-length code", *Electronics Letters*, vol. 29, pp. 1741-1742, Sept. 1993.
- [12] Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizers design", *IEEE Trans. on Communications*, vol. COM-28, pp. 84-95, Jan. 1980.
- [13] J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and video", *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Processing*, 1991, pp.2661-2664.
- [14] Liu and A. Zaccarin, "New fast algorithms for the estimation of block motion vectors", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 3, pp. 148-157, Apr. 1993.
- [15] N. Netravali and B. G. Haskell, *Digital Pictures - Representation and Compression*. New York: Plenum Press, 1988.
- [16] ITU-T Recommendation H.261, "Video codec for audiovisual services at  $p \times 64$  kbit/s, Mar. 1993.
- [17] Liou, "Visual telephone as an ISDN application," *IEEE Communications Magazine*, vol. 28, pp. 30-46, Feb. 1990.
- [18] Li, A. Lundmark and R. Forchheimer, "Image sequence coding at very low bitrates: a review," *IEEE Trans. on Image Processing*, vol. 3, pp. 589-609, Sept. 1994.
- [19] V. Lamar, J. C. M. Bermudez and R. Seara, "Interframe image coding using quadtree for motion estimation," *38<sup>th</sup> IEEE Midwest Symposium on Circuits and Systems*, Aug. 1995.

**Marcus Vinicius Lamar** nasceu em Porto Alegre em 12 de abril de 1970. Graduiu-se em Engenharia Elétrica pela Universidade Federal do Rio Grande do Sul (UFRGS) em 1992. Atualmente está em fase de conclusão do Mestrado em Engenharia Elétrica pela Universidade Federal de Santa Catarina (UFSC) na área de Circuitos e Instrumentação Eletrônica. É Professor Auxiliar do Departamento de Engenharia Elétrica da Universidade Federal do Paraná (UFPR), onde desenvolve atividades de pesquisa e ensino nas áreas de Processamento Digital de Sinais, Codificação de Imagens e Vídeo e Reconhecimento de Voz.

**José Carlos Moreira Bermudez** nasceu no Rio de Janeiro, RJ, em 6 de março de 1956. Graduiu-se em Engenharia Elétrica pela Universidade Federal do Rio de Janeiro (UFRJ), Rio de Janeiro, RJ, em 1978. Obteve o título de Mestre em Ciências em Engenharia Elétrica pela UFRJ. Obteve o título de Doutor em Engenharia Elétrica pela Concordia University, Montreal, Canadá, em 1985. Ingressou no Departamento de Engenharia Elétrica da UFSC em 1985, onde atualmente é Professor Titular e desenvolve suas atividades de pesquisa e ensino nas áreas de: Processamento Digital de Sinais e Processamento Adaptativo de Sinais. É Editor Associado da *IEEE Transactions on Signal Processing* na área de filtragem adaptativa.

**Rui Seara** nasceu em Florianópolis, SC, em 31 de Agosto de 1951. Graduiu-se em Engenharia Elétrica pela Universidade Federal de Santa Catarina (UFSC), Florianópolis, SC, em 1975. Obteve o título de Mestre em Ciências em Engenharia Elétrica pela UFSC, em 1980. Especializou-se em Instrumentação-Metrologia pela Ecole Supérieure d'Electricité de Paris, França, em 1982. Obteve o título de Doutor em Engenharia Elétrica pela Université Paris Sud de Paris, França, em 1984. Ingressou no Departamento de Engenharia Elétrica da UFSC em 1976, onde atualmente é Professor Titular e desenvolve suas atividades de pesquisa e ensino nas áreas de: Filtros Adaptáveis, Filtros Analógicos, Filtros Digitais, Processamento de Imagens, Processamento de Voz e Instrumentação Eletrônica.