

Um Esquema de Compressão de Sequências de Imagens Baseado na DCT

CARLOS ANTONIO REINALDO COSTA¹
PAULO LÍCIO DE GEUS²

¹DCC - IMECC - UNICAMP
Caixa Postal 6065
13081 Campinas, SP, Brasil
ccosta@dcc.unicamp.br

²DCC - IMECC - UNICAMP
Caixa Postal 6065
13081 Campinas, SP, Brasil
paulo@dcc.unicamp.br

Abstract. The growth of modern telecommunications technology have made possible the use of digital video transmission in several types of applications. Digital transmission has many advantages over analog transmission; however, there are some problems that appear with the digital transmission of image sequences: the digitalization process increases bandwidth requirements and most local area networks do not support the cost of digital video transmission in real-time. Image compression techniques seek to minimize this cost; they must be relatively fast and, at the same time, able to reach high compression rates. In this paper, it is presented a new scheme for image sequence compression that combines the discrete cosine transform with movement detection and progressive transmission strategies. Detailed descriptions of the method and its variants are given, and some experimental results are analysed.

1 Introdução

As técnicas de compressão de imagem surgem da necessidade de minimizar o número de bits necessários à representação de imagens para armazenamento ou transmissão. Elas podem ser divididas em dois tipos: as que envolvem perda de informação e as que preservam informação. Obviamente, quando pode existir perda de informação as taxas de compressão atingidas são muito maiores; entretanto, em algumas aplicações a perda de informação não é admissível e raramente se obtém taxas de compressão superiores a 2:1 para imagens de mundo real.

Alguns critérios de fidelidade, subjetivos ou quantitativos, podem ser usados para determinar até que ponto a informação pode ser perdida. Critérios subjetivos são baseados na inspeção visual das imagens por parte de um grupo de observadores, e critérios quantitativos utilizam algumas expressões matemáticas que medem a distorção das imagens reconstruídas em relação às imagens originais.

Uma grande variedade de técnicas de compressão de imagens tem sido desenvolvida nos últimos anos, algumas delas voltadas para imagens isoladas e outras para seqüências. Na pesquisa aqui realizada, algumas destas técnicas foram combinadas en-

tre si para explorar tanto a redundância espacial (dentro de um mesmo quadro) quanto a redundância temporal (entre quadros contíguos) com o objetivo de atingir taxas relativamente altas de compressão. A transformada do cosseno, ou simplesmente DCT (*Discrete Cosine Transform*), foi usada para explorar a redundância espacial. A escolha desta técnica deve-se ao seu bom desempenho em compressão de imagens, apesar de seu alto custo computacional. Com os avanços recentes da tecnologia digital já existem chips que implementam DCT para aplicações de tempo real. Para explorar a redundância temporal, foram usadas estratégias de detecção de movimento e transmissão progressiva: apenas blocos detectados com movimento são codificados, com os demais sendo simplesmente copiados do quadro anterior.

Iniciamos com um breve resumo sobre codificação por transformadas, seguindo-se o esquema desenvolvido e algumas de suas variantes, terminando com uma análise de resultados experimentais.

2 Codificação por Transformadas

Existem algumas transformações lineares, normalmente chamadas de transformadas, que são frequentemente usadas em processamento de imagens para

diversos propósitos, como por exemplo detecção de arestas e filtragem de ruído.

Para imagens de mundo real, a maioria destas transformadas tende a concentrar a maior parte da energia em uma pequena quantidade de coeficientes, fato este que pode ser explorado na elaboração de métodos de compressão de imagens. Alocando-se seletivamente um maior número de bits para coeficientes com maior concentração de energia, pode-se obter boas taxas de compressão sem uma significativa degradação das imagens reconstruídas.

As transformadas usadas em processamento de imagens podem ser divididas em dois grupos: senoidais e não senoidais. As transformadas de Fourier, do cosseno e do seno são exemplos de transformadas senoidais; as transformadas de Hadamard, Haar e Slant são exemplos de transformadas não senoidais [Pra78]. Além destas existe também a transformada de Karhunen-Loeve (KLT) que é ótima no sentido de concentrar o máximo de energia em um mínimo de coeficientes. A KLT, entretanto, não é utilizada na prática devido ao seu alto custo computacional: não possui algoritmo rápido e depende de algumas características estatísticas que normalmente não são representáveis de forma analítica. Na prática a KLT pode ser substituída por outras transformadas que possuem algoritmo rápido e que são capazes de aproximar seu desempenho. A seguir damos a definição de duas das transformadas mais amplamente usadas, considerando imagens de tamanho $N \times N$:

Transformada de Fourier Discreta (DFT)

$$U(k, l) = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} u(m, n) W_N^{km+ln}$$

$$u(m, n) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} U(k, l) W_N^{-km-ln}$$

onde,

$$W_N = \exp\left(\frac{-2\pi i}{N}\right) \text{ e } i = \sqrt{-1}$$

Transformada do Cosseno Discreta (DCT)

$$U(k, l) = \frac{2}{N} c(k)c(l) \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} u(m, n) C_N(m, n; k, l)$$

$$u(m, n) = \frac{2}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} c(k)c(l) U(k, l) C_N(m, n; k, l)$$

onde,

$$C_N(m, n; k, l) = \cos\left(\frac{\pi(2m+1)k}{2N}\right) \cos\left(\frac{\pi(2n+1)l}{2N}\right)$$

$$c(0) = \frac{1}{\sqrt{2}}$$

$$c(j) = 1 \text{ para } j \neq 0$$

Foi provado experimentalmente que a DCT é a transformada com algoritmo rápido que melhor aproxima o desempenho da KLT [Jai89], sendo portanto a mais utilizada em compressão de imagens. O coeficiente $U(0, 0)$ de uma imagem transformada por DCT é normalmente chamado de nível DC e seu valor é proporcional ao brilho médio da imagem; os demais coeficientes, chamados de níveis AC, representam as componentes de frequência espacial, o que significa que a DCT dá uma representação da imagem no domínio da frequência, enquanto que a imagem original é uma representação no domínio do espaço. Para imagens de mundo real, a maior parte da energia tende a ficar concentrada nas proximidades do nível DC, onde estão os coeficientes de baixa frequência espacial. A compressão pode ser obtida alocando-se um número de bits proporcional à concentração de energia em cada coeficiente. A variância ou desvio padrão dos coeficientes da imagem transformada podem ser usados para estimar a distribuição de energia. Por razões práticas, a DCT não é aplicada sobre a imagem como um todo; ao invés disto, ela é dividida em pequenos blocos retangulares com tamanhos que normalmente estão entre 8×8 e 32×32 . A Figura 1 mostra uma típica alocação de bits para DCT de blocos de tamanho 16×16 mantendo uma média de 1 bit/pixel.

8	7	6	5	3	3	2	2	2	1	1	1	1	1	0	0
7	6	5	4	3	3	2	2	1	1	1	1	1	1	0	0
6	5	4	3	3	2	2	2	1	1	1	1	1	1	0	0
5	4	3	3	3	2	2	2	1	1	1	1	1	1	0	0
3	3	3	3	2	2	2	2	1	1	1	1	1	1	0	0
3	3	2	2	2	2	2	2	1	1	1	1	1	1	0	0
2	2	2	2	2	2	2	1	1	1	1	1	1	1	0	0
2	2	2	2	1	1	1	1	1	1	1	1	1	1	0	0
2	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figura 1: Exemplo de alocação de bits para DCT.

Além da alocação de bits, outro ponto essencial para um bom desempenho da DCT é a quan-

tização dos coeficientes. No caso ilustrado na Figura 1, oito quantizadores diferentes são necessários, um para cada quantidade de bits indicada. Bons resultados podem ser obtidos usando-se quantizadores de Lloyd-Max, que minimizam o erro médio quadrático para um número fixo de níveis [Jai89].

Apesar de existirem algoritmos rápidos (ver por exemplo [Mak80] e [CL91]), o custo computacional da DCT ainda é muito alto para algumas aplicações; entretanto, a recente implementação VLSI já permite seu uso em aplicações de tempo real.

3 O Esquema Proposto

Nesta seção apresentamos um esquema de compressão de imagens baseado na DCT que combina estratégias de detecção de movimento e transmissão progressiva. A combinação destas duas estratégias foi inicialmente proposta por Geus [dG90] que elaborou um esquema baseado em quantização vetorial chamado MDPT/VQ (*Movement Detected, Progressive Transmission with Vector Quantization*). O esquema descrito aqui, por usar estratégias semelhantes, foi chamado de MDPT/DCT; a principal diferença entre estes métodos é a substituição da quantização vetorial pela DCT. A seguir descrevemos os elementos básicos do método.

Detecção de Movimento. As imagens são divididas em blocos de 8×8 , sobre os quais é realizada detecção de movimento, individualmente. Blocos sem movimento não são codificados, sendo mantidos do quadro anterior no processo de decodificação. Isto minimiza o número de blocos a ser codificado sem causar um prejuízo significativo na qualidade das imagens. Bons resultados são obtidos medindo-se a distorção entre blocos na mesma posição em quadros subsequentes e verificando-se se está abaixo ou acima de um limiar convenientemente escolhido. Este limiar é usado para evitar que blocos sejam detectados com movimento apenas por oscilação de ruído temporal. A medida de distorção utilizada foi o erro quadrático, definido para dois vetores $\mathbf{v} = (v_0, \dots, v_{k-1})$ e $\mathbf{w} = (w_0, \dots, w_{k-1})$ como,

$$d(\mathbf{v}, \mathbf{w}) = \sum_{i=0}^{k-1} (v_i - w_i)^2$$

Descrição Prévia do Bloco. Uma vez detectado movimento em um bloco, um método eficiente deve ser usado para dar uma primeira descrição do mesmo. Várias técnicas podem ser usadas com este objetivo, como por exemplo, BTC (*Block Truncation Coding*), subamostragem com interpolação, quantização vetorial e codificação por transformadas [dG90]. Entre

estas, as duas últimas são as que produzem os melhores resultados. Geus utilizou quantização vetorial; aqui, usamos codificação por transformadas (DCT).

Transmissão Progressiva. Esta estratégia é baseada em um trabalho de Dreizen [Dre87] e consiste em transmitir a informação de atualização ao longo de mais de um quadro da sequência de imagens. Após a descrição prévia do bloco, informação adicional continua a ser transmitida até que a distorção esteja abaixo de um limiar predefinido (para a medida de distorção usou-se novamente o erro quadrático). A DCT é mais facilmente adaptável à estratégia de transmissão progressiva que a quantização vetorial, permitindo o desenvolvimento de um método muito mais simples e flexível que o MDPT/VQ.

A idéia básica do método é fazer uma alocação de bits para blocos de 8×8 e transmitir os coeficientes ao longo de mais de um quadro. Para a descrição prévia do bloco seriam transmitidos o nível DC e mais alguns coeficientes próximos (baixas frequências espaciais). Nos quadros seguintes são transmitidos os coeficientes que representam frequências espaciais maiores. Entretanto, existe uma condição a ser satisfeita antes da transmissão da informação referente à descrição prévia do bloco, ou a qualquer dos passos de transmissão progressiva: a informação só é transmitida se ela for capaz de produzir uma queda significativa na distorção do bloco com relação à imagem original. Pode acontecer, com alguma frequência, que o bloco reconstruído em certo estágio de transmissão não representa uma melhora significativa em relação à representação corrente do mesmo. Pode mesmo apresentar uma distorção maior. Isto acontece, por exemplo, quando é detectado um movimento muito suave. Para evitar uma possível transmissão desnecessária de informação, a distorção do bloco reconstruído é comparada com a distorção da representação corrente do bloco; a informação só é transmitida se houver uma queda significativa na distorção.

O método tem como entrada, além das imagens originais, uma alocação de bits dividida em n regiões r_i ($i = 0, \dots, n - 1$). O esquema de transmissão progressiva pode ser visto como uma máquina de estados finita cuja função de transição de estados depende do estado corrente e da distorção do bloco reconstruído. No momento em que é detectado movimento no bloco (estado inicial e_0), faz-se sua descrição prévia, quantizando-se os coeficientes indicados na região r_0 e substituindo os restantes por zeros; a distorção do bloco reconstruído é medida e subtraída da distorção corrente; se a diferença for

maior que um limiar L_1 a informação é transmitida, caso contrário não há transmissão. Se for detectado novo movimento no bloco a máquina volta ao estado inicial e_0 ; caso contrário passamos ao estado e_1 , a partir do qual o processamento e a transição de estados ocorre da seguinte maneira: supondo-se que estamos em um estado e_i tal que $1 \leq i \leq n-1$,

1. Mede a distorção corrente D_c e compara com o limiar L_1 .
2. Se $D_c > L_1$: reconstrói o bloco atualizando os coeficientes indicados na região r_i ; calcula nova distorção D_r ; se $D_c - D_r > L_1$ transmite informação, caso contrário não há transmissão; estado passa a ser e_{i+1} .
3. Se $D_c \leq L_1$: não há processamento; estado passa a ser e_n .

Se estamos em um estado e_j tal que $n \leq j \leq 2n-1$, o processamento é similar, mas com algumas diferenças:

1. Mede a distorção corrente D_c e compara com um limiar L_2 ($< L_1$).
2. Se $D_c > L_2$: reconstrói o bloco atualizando os coeficientes indicados na região r_{j-n} ; calcula nova distorção D_r ; se $D_c - D_r > L_2$ transmite informação, caso contrário não há transmissão; estado passa a ser e_{j+1} ; se $j+1 = 2n$ pára (e_{2n} = estado final).
3. Se $D_c \leq L_2$: não há processamento; estado passa a ser e_{2n} e pára.

A região r_0 contém o nível DC e outros coeficientes de mais baixa frequência espacial, as regiões seguintes contém coeficientes correspondentes a frequências progressivamente maiores. O uso de dois limiares diferentes, sendo o primeiro maior que o segundo, faz com que sejam atualizados em primeiro lugar apenas os coeficientes das regiões que mais influenciam na qualidade do bloco.

Neste método, além dos dois *buffers* com imagens originais para detecção de movimento e de um *buffer* com a imagem reconstruída, temos também um *buffer* que contém os coeficientes reconstruídos da imagem transformada. Na descrição prévia do bloco (e_0) são atualizados os coeficientes indicados na região r_0 e os demais são substituídos por zeros; para os demais estados são atualizados os coeficientes correspondentes, mantendo-se os valores dos já transmitidos. Caso não seja transmitida a informação de atualização, todos os coeficientes permanecem inalterados e o bloco é tratado como inativo.

Um esquema baseado em *run-length* é usado para codificar os blocos inativos: apenas os tamanhos dos vãos de blocos inativos são codificados. A Figura 2 mostra uma possível estrutura de dados para o código gerado pelo MDPT/DCT. Os comprimentos dos vãos de blocos inativos são codificados com 3 bits se forem menores que 8, caso contrário são codificados com 8 bits; vãos maiores que 255 são divididos em 2 ou mais vãos; o estado do bloco indica qual região deve ser atualizada e é codificado com $\lceil \log_2 n + 1 \rceil$ bits, onde n é o número de regiões. Esta codificação certamente não é ótima: técnicas de codificação estatística sem perda poderiam ser usadas para minimizar o volume de dados; entretanto, esta estrutura tem a vantagem de ser bastante simples e fácil de implementar, além de termos obtido resultados bastante satisfatórios com este esquema; mais adiante mostramos alguns destes resultados, que foram obtidos com diferentes versões de MDPT/DCT.

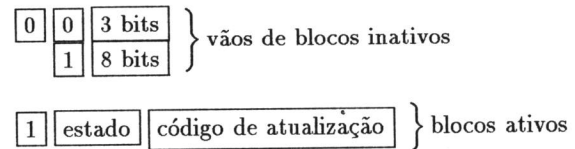


Figura 2: Código gerado para MDPT/DCT.

3.1 Variações

A qualidade das imagens reproduzidas e as taxas de compressão atingidas com o MDPT/DCT são diretamente ligadas à média de bits alocados por coeficiente e ao número de regiões em que o bloco transformado é dividido. Da média de bits por coeficiente depende a qualidade final do bloco, após todo o processo de transmissão progressiva. Do número de regiões depende a qualidade do bloco durante os passos intermediários. A taxa de compressão depende de ambos os fatores; particularmente, à medida que se diminui o número de bits para cada passo, também diminui o número de blocos para os quais a informação é transmitida, uma vez que a qualidade da reprodução tende a ser inferior, e isto faz com que a taxa de compressão aumente ainda mais.

A alocação de bits e as regiões processadas em cada passo podem ser determinadas pela própria aplicação, de acordo com suas necessidades, ou escolhidas dentro de um conjunto predefinido contendo diferentes variações. Isto faz com que este método tenha uma flexibilidade muito maior que o MDPT/VQ.

(r_0)	8	7	6	0	0	0	0	0	0
	7	0	0	0	0	0	0	0	0
	6	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
(r_1)	0	0	0	5	4	0	0	0	0
	0	5	4	0	0	0	0	0	0
	0	4	0	0	0	0	0	0	0
	5	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
(r_2)	0	0	0	0	0	3	2	0	0
	0	0	0	3	2	0	0	0	0
	0	0	3	2	2	0	0	0	0
	0	3	2	2	0	0	0	0	0
	0	2	2	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
(r_3)	0	0	0	0	0	0	0	0	2
	0	0	0	0	0	2	1	1	
	0	0	0	0	0	1	1	0	
	0	0	0	0	2	1	1	0	
	0	0	0	2	1	1	1	0	
	0	2	1	1	1	1	0	0	
	0	1	1	1	1	0	0	0	
	2	1	0	0	0	0	0	0	

Figura 3: Alocações de bits para Versão 2/4.

Para demonstrar esta flexibilidade elaboramos algumas variações do MDPT/DCT, para diferentes taxas de compressão e níveis de qualidade. O procedimento é mantido para todas as variações; muda-se apenas a média de bits por coeficiente e o número de regiões de alocação. Os detalhes de como foram feitas estas alocações e a quantização dos coeficientes são dados em [Cos93]. Para facilitar a exposição, vamos chamar a versão do MDPT/DCT que usa uma alocação de b bits por coeficiente dividida em r regiões de Versão b/r .

Usamos três variações em nossos testes: a Versão 2/4, cuja alocação é mostrada na Figura 3, destina-se a aplicações onde há necessidade de se atingir taxas elevadas de compressão (superiores a 40:1) mas

admitindo certa degradação da imagem; a Versão 3/3, que usa a alocação mostrada na Figura 4, onde se preteriu altas taxas de compressão em favor de uma maior qualidade nas imagens reproduzidas; e a Versão 3/4, intermediária, que usa a alocação que é mostrada na Figura 5.

(r_0)	8	8	7	6	0	0	0	0	0
	8	6	5	0	0	0	0	0	0
	7	5	0	0	0	0	0	0	0
	6	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
(r_1)	0	0	0	0	5	4	3	0	0
	0	0	0	4	3	3	0	0	0
	0	0	4	3	3	0	0	0	0
	0	4	3	3	0	0	0	0	0
	5	3	3	0	0	0	0	0	0
	4	3	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0
(r_2)	0	0	0	0	0	0	0	0	3
	0	0	0	0	0	0	0	2	2
	0	0	0	0	0	0	2	2	2
	0	0	0	0	2	2	2	2	2
	0	0	0	2	2	2	2	2	1
	0	0	2	2	2	2	2	1	1
	0	2	2	2	2	1	1	1	1
	3	2	2	1	1	1	1	1	1

Figura 4: Alocações de bits para Versão 3/3.

3.2 Resultados

Foram processadas as mesmas seqüências usadas para testar o MDPT/VQ, e obtendo-se resultados significativamente superiores. São seqüências com 64 quadros de 256×256 pixels. A seqüência *hand3* contém bastante movimento e testa a reconstrução de fundo após a passagem de um objeto em movimento. A seqüência *talk2* é uma cena típica de teleconferência, com movimentos moderados (rosto e ombros).

3.2.1 Versão 2/4

A Tabela 1 sumariza os resultados obtidos quanto à taxa de compressão e relação sinal/ruído para as seqüências *hand3* e *talk2* codificadas pela Versão 2/4; A Figura 6 mostra os quadros 31 e 63 da seqüência *hand3* reconstruída, assim como suas imagens de erro.

(r_0)	8	8	7	6	0	0	0	0
	8	6	0	0	0	0	0	0
	7	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
(r_1)	0	0	0	0	5	4	0	0
	0	0	5	4	0	0	0	0
	0	5	4	0	0	0	0	0
	6	4	0	0	0	0	0	0
	5	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
(r_2)	0	0	0	0	0	0	3	3
	0	0	0	0	3	3	2	2
	0	0	0	3	3	2	0	0
	0	0	3	3	0	0	0	0
	0	3	3	0	0	0	0	0
	0	3	0	0	0	0	0	0
	3	2	0	0	0	0	0	0
	3	2	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
(r_3)	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	2	2
	0	0	0	0	2	2	2	2
	0	0	0	2	2	2	2	1
	0	0	2	2	2	2	1	1
	0	0	2	2	2	1	1	1
	0	0	2	1	1	1	1	1
	0	0	2	1	1	1	1	1
	0	0	2	1	1	1	1	1

Figura 5: Alocações de bits para Versão 3/4.

Os principais defeitos visíveis nas imagens concentram-se nas arestas em movimento ou em regiões com muitos detalhes, onde muitas vezes é possível visualizar as fronteiras entre os blocos. Olhando-se mais atentamente pode-se também perceber alguns erros em regiões sem movimento que foram reconstruídas pelo processo de transmissão progressiva; entretanto estas falhas são pequenas e pouco interferem na qualidade da imagem. Maiores requisitos de qualidade podem ser atendidos por uma combinação de parâmetros mais próxima às das outras versões apresentadas a seguir.

Considerando-se que seqüências de imagens são dinâmicas, esses artefatos não chegam a perturbar seriamente o espectador, especialmente quando con-

Seqüência	Taxa Média	
	Compressão	SNR (dB)
<i>hand3</i>	48,00:1	28,81
<i>talk2</i>	101,01:1	30,95

Tabela 1: Resultados obtidos com a Versão 2/4.

sideramos a elevada taxa de compressão atingida. O resultado final pode ser considerado satisfatório para várias aplicações.

3.2.2 Versão 3/3

Nesta versão a média de bits alocados por coeficiente foi aumentada de 2 para 3 e o número de regiões foi diminuído de 4 para 3. Como resultado a quantidade média de bits processados em cada passo de transmissão progressiva dobrou em relação à versão anterior. Isto resultou em uma melhora bastante significativa na qualidade da imagem, porém ao custo de queda significativa na taxa de compressão. Os resultados obtidos estão sumarizados na Tabela 2. Os quadros 31 e 63 da seqüência *hand3* são mostrados na Figura 7 junto com as imagens de erro.

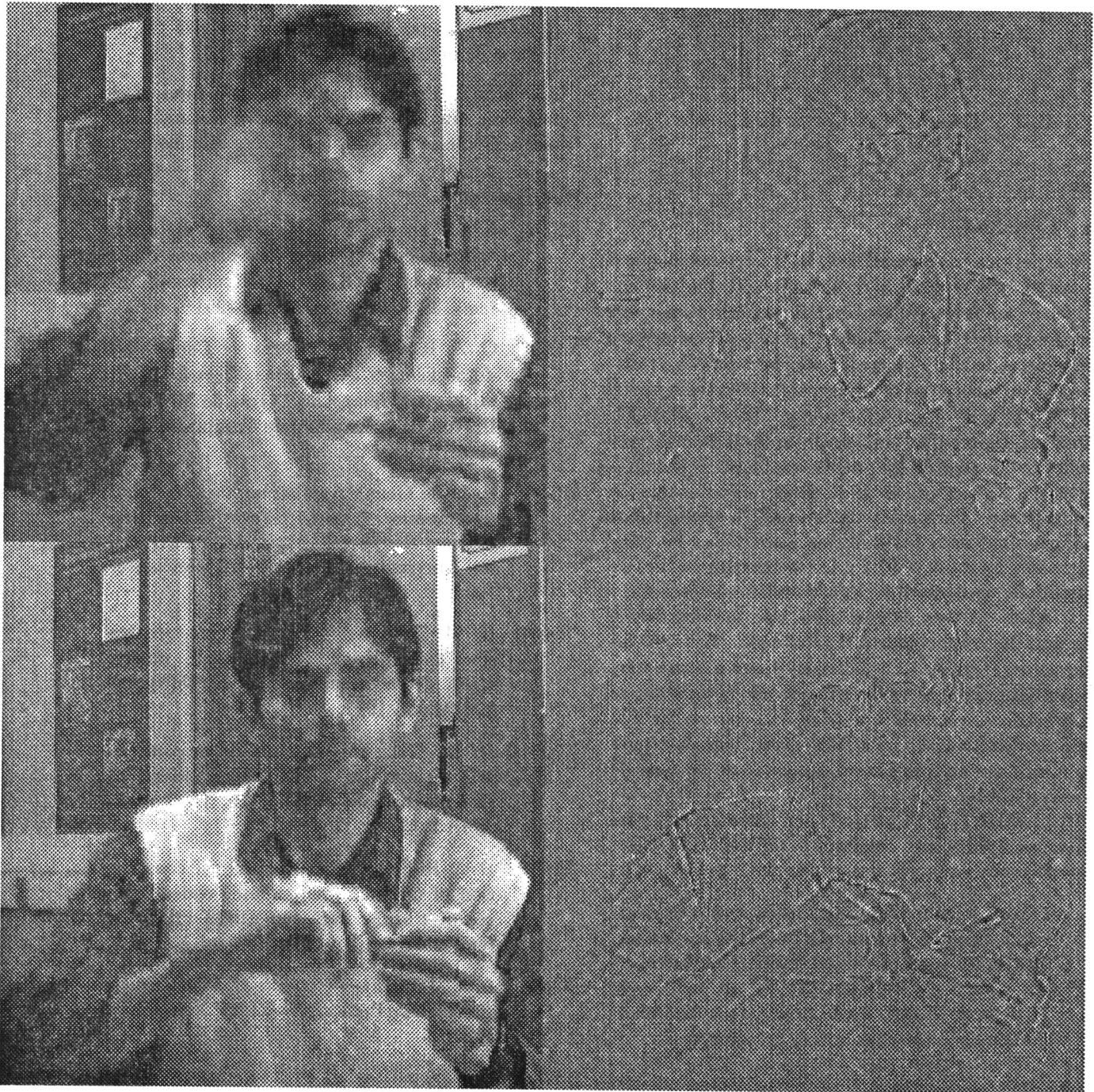
Seqüência	Taxa Média	
	Compressão	SNR (dB)
<i>hand3</i>	17,39:1	30,91
<i>talk2</i>	29,22:1	32,42

Tabela 2: Resultados obtidos com a Versão 3/3.

Algumas falhas ainda permanecem nas imagens reproduzidas, principalmente nas arestas em movimento, porém são pouco visíveis sob observação dinâmica. O esquema de transmissão progressiva reproduziu os blocos sem movimento de maneira bem mais fiel e as fronteiras entre os blocos tornaram-se bem menos visíveis. As taxas de compressão que atingimos, apesar de serem bem menores que as da Versão 2/4, ainda são satisfatórias para muitos casos, e a qualidade das imagens pode ser considerada boa e suficiente para um grande número de aplicações.

3.2.3 Versão 3/4

Esta Versão demonstra novamente a flexibilidade do MDPT/DCT; ela é intermediária entre as outras duas e permite atingir níveis moderados de compressão e qualidade. A Tabela 3 mostra seu desempenho e a Figura 8 mostra os quadros 31 e 63 da seqüência *hand3* reproduzida, com suas imagens de erro.



SNR: quadro 31 - 28,91 dB; quadro 63 - 28,64 dB.

Figura 6: Seqüência *hand3* codificada pela Versão 2/4.

Seqüência	Taxa Média	
	Compressão	SNR (dB)
<i>hand3</i>	27,45:1	29,91
<i>talk2</i>	48,13:1	31,74

Tabela 3: Resultados obtidos com a Versão 3/4.

4 Conclusão

Os experimentos com MDPT/DCT mostraram que é possível atingir resultados bastante satisfatórios em diversos tipos de aplicação. Foi possível atingir um elevado padrão de qualidade com taxas de compressão superiores a 12:1, o que é suficiente para a transmissão de seqüências de imagens de vídeo (com padrão de TV) através de redes locais como *Ethernet* [dG90]. Quando é necessário transmitir imagens de maior resolução, ou quando há outras transmissões simultâneas, há a necessidade de se obter maiores taxas de compressão. Para estes casos também conseguimos alguns resultados satisfatórios, com taxas de compressão superiores a 50:1 e mantendo um padrão de qualidade ainda aceitável para muitas aplicações.

Além do bom desempenho, este método dispõe de grande simplicidade e flexibilidade. Pode ser facilmente integrável a aplicações, o que o torna uma boa alternativa para sistemas que não necessitam de codificadores muito sofisticados, tais como o MPEG, que foi desenvolvido pela ISO¹ especificamente para se tornar um padrão para compressão de imagens de vídeo [Gal91].

O MDPT/DCT já pode ser bastante útil para diversas aplicações; entretanto ele ainda pode ser aperfeiçoado em alguns aspectos, como por exemplo:

- O tipo de código que sugerimos para o esquema não é ótimo e certamente pode ser melhorado. Utilizando técnicas de codificação estatística, como código de Huffman [Huf52] ou codificação aritmética [RJ79], ainda podemos aumentar de maneira significativa a taxa de compressão.
- Toda a codificação do MDPT baseia-se em reconstruções a partir de uma única imagem base que corresponde à primeira imagem da seqüência. Isto pode causar problemas como propagação de erros e dificultar o acesso a imagens em posições aleatórias da seqüência. Em um sistema mais completo a reconstrução deveria ser possível a partir de várias imagens base distribuídas entre

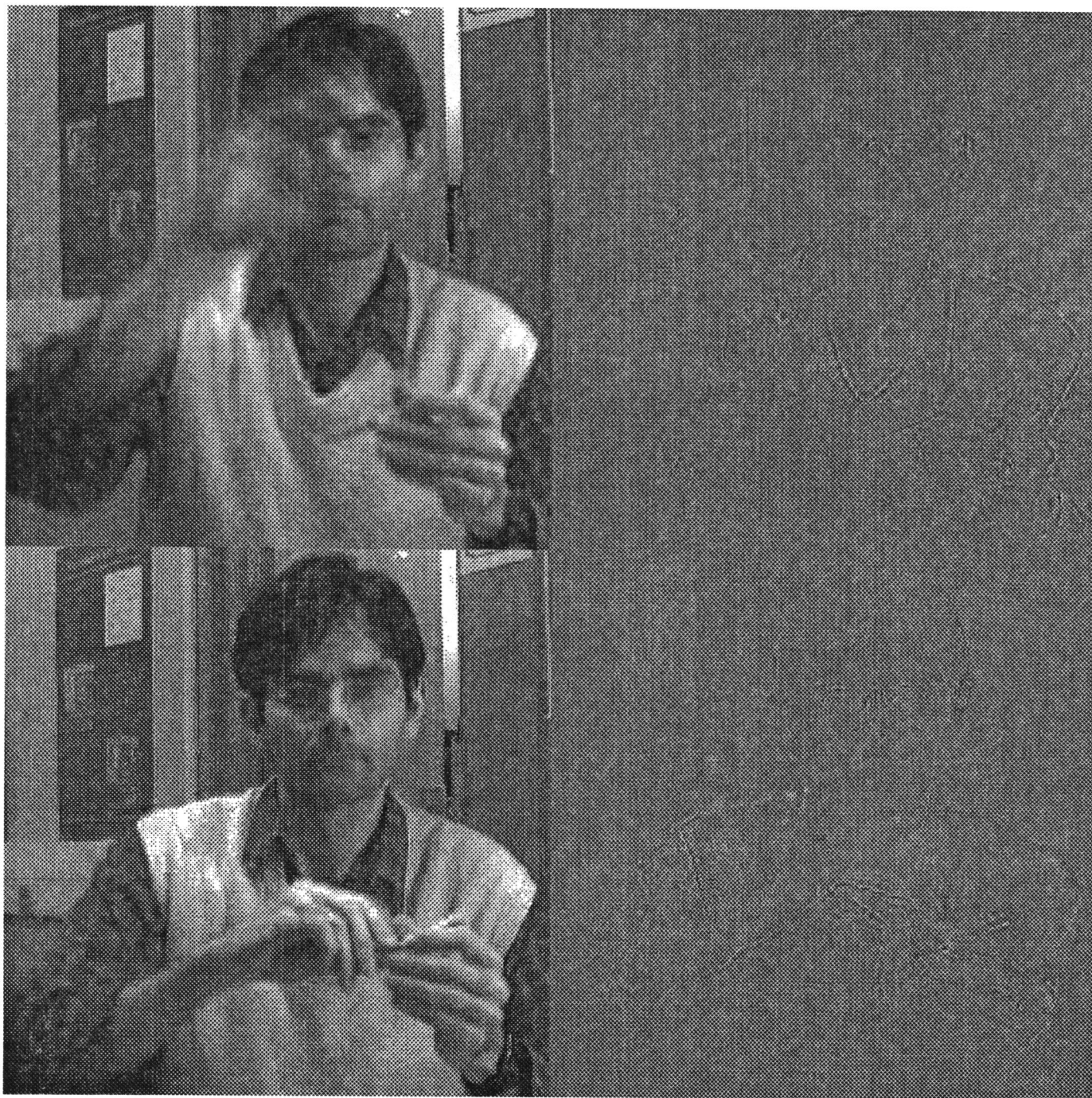
intervalos regulares ao longo da seqüência. Para estas imagens, deveria ser desenvolvido um esquema eficiente de compressão que não dependesse de outras imagens da seqüência.

O MDPT/DCT, como descrevemos aqui, pode dar início a uma alternativa simples e eficiente para transmissão de seqüências de imagens digitais.

Referências

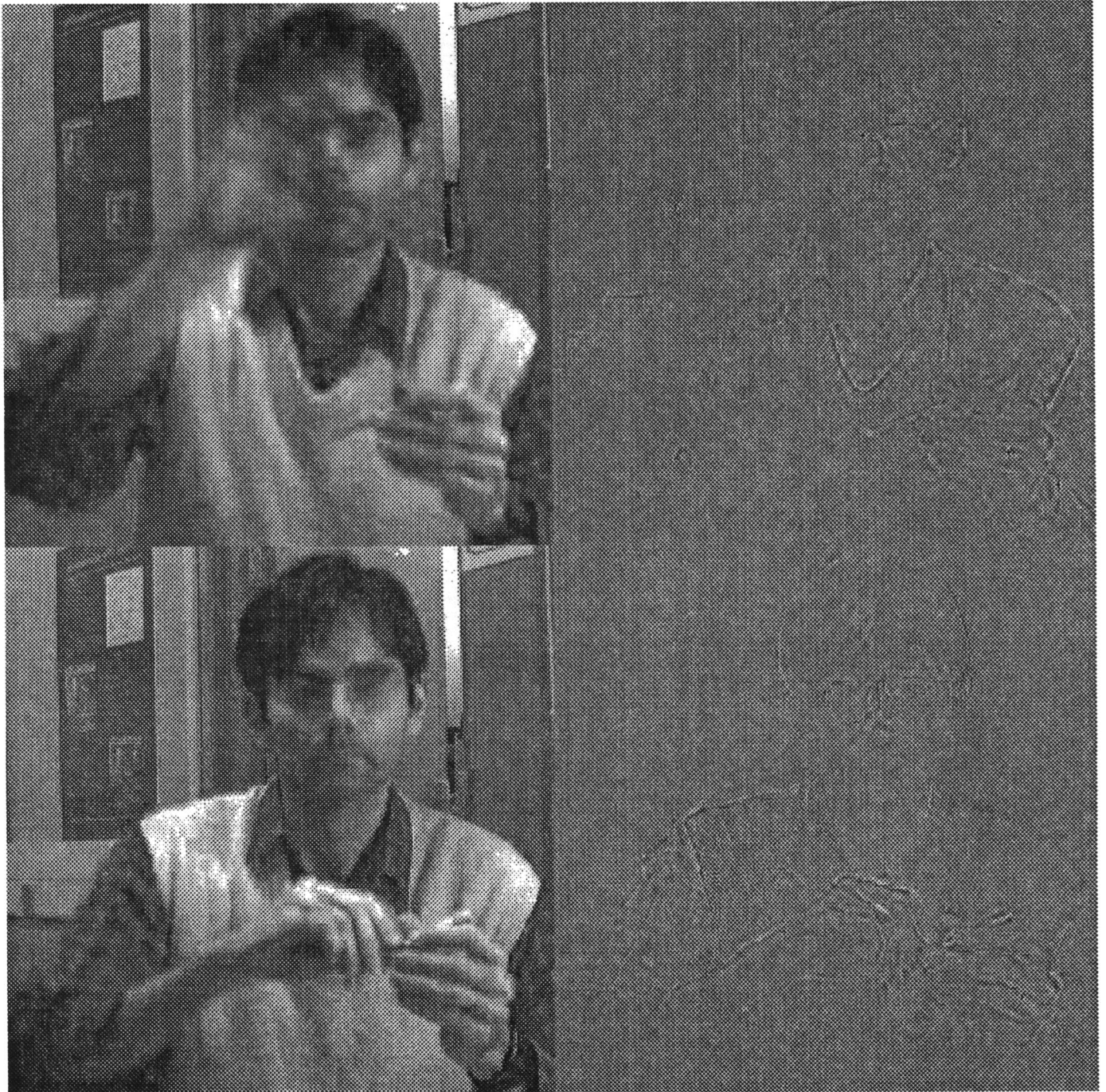
- [CL91] N. I. Cho and S. U. Lee. Fast algorithm and implementation of 2-D discrete cosine transform. *IEEE Trans. Circuits and Systems*, 38(3):297–305, March 1991.
- [Cos93] C. A. R. Costa. *Técnicas de Compressão de Seqüências de Imagens Visando Transmissão em Tempo Real*. Tese de Mestrado, Universidade Estadual de Campinas, Campinas, 1993.
- [dG90] P. L. de Geus. *Approaches to Live Image Transmission Between Workstations Over Limited-Bandwidth Networks*. PhD thesis, University of Manchester, Manchester, 1990.
- [Dre87] H. M. Dreizen. Content-driven progressive transmission of grey-scale images. *IEEE Trans. Communications*, 35(3):289–296, March 1987.
- [Gal91] D. Le Gal. MPEG: A video compression standard for multimedia applications. *Communications of the ACM*, 34(4):46–58, April 1991.
- [Huf52] D. A. Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101, September 1952.
- [Jai89] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall Inc., New Jersey, 1989.
- [Mak80] J. Makhoul. A fast cosine transform in one and two dimensions. *IEEE Trans. Acoust., Speech and Signal Proc.*, 28(1):27–34, February 1980.
- [Pra78] W. K. Pratt. *Digital Image Processing*. John Wiley & Sons, New York, 1978.
- [RJ79] J. Rissanen and G. G. Langdon Jr. Arithmetic coding. *IBM J. Res. Develop.*, 23(2):149–162, March 1979.

¹ Acrônimo de *International Standards Organization* apesar do nome verdadeiro ser *International Organization for Standardization*.



SNR: quadro 31 - 31,26 dB; quadro 63 - 30,80 dB.

Figura 7: Seqüência *hand3* codificada pela Versão 3/3.



SNR: quadro 31 - 30,20 dB; quadro 63 - 29,82 dB.

Figura 8: Seqüência *hand3* codificada pela Versão 3/4.