

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA
DEPARTAMENTO DE COMUNICAÇÕES

SIMULAÇÕES PARA A CODIFICAÇÃO DE IMAGENS PARADAS USANDO O PADRÃO JPEG BÁSICO.

Este exemplar contém a versão final da tese
defendida por Fernando Chung Joo universidade
Jugadora em 03/Out/95
Yuzo Iano orientador

Autor: Fernando Chung Joo
Orientador: Prof. Dr. Yuzo Iano

Dissertação apresentada à Faculdade de Engenharia Elétrica da Universidade Estadual de Campinas, como parte dos requisitos para a obtenção do Título de Mestre em Engenharia Elétrica.

Outubro 1995



| | |
|----------------------------|---------------------------------------|
| UNIDADE | 78C |
| M. C. C. T. A. D. A. : | T/UNICAMP |
| | C472s |
| V. _____ | Ex. _____ |
| TI M80 BC/ | 27609 |
| PROD. | 667196 |
| C <input type="checkbox"/> | D <input checked="" type="checkbox"/> |
| PREÇO | R\$ 11,00 |
| DATA | 03/05/96 |
| N.º CPD | |

CM-00087587-0

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

C472s Chung Joo, Fernando
Simulações para codificação de imagens paradas usando o padrão JPEG básico / Fernando Chung Joo.--
Campinas, SP: [s.n.], 1995.

Orientador: Yuzo Iano.
Dissertação (mestrado) - Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica.

1. Compressão de dados (Telecomunicações).
2. Codificação . 3. Processamento de imagens. I. Iano,
Yuzo. II. Universidade Estadual de Campinas. Faculdade
de Engenharia Elétrica. III. Título.

AGRADECIMENTOS

Ao Prof. Dr. Yuzo Iano, pela amizade, apoio e orientação deste trabalho.

Aos meus pais Juana e Jorge; às minhas irmãs Beatriz e Jessica; e à minha sobrinha Lorenita pelo carinho, apoio e incentivo.

Aos Professores do DECOM / FEE, com os quais tive contato.

Aos colegas do curso de Mestrado, pelo apoio.

Aos amigos Rodrigo, Miguel, Ayres, Fançony, Eddy, Ruth, Richard e Edgard pela colaboração e amizade.

Aos amigos Eng. Gustavo Rosello e Cap. Aldo Bravo pelo incentivo e confiança.

Aos funcionários da FEE, pelo auxílio.

A CAPES e ao Governo Brasileiro pelo apoio financeiro e oportunidade oferecida.

Às pessoas que, de alguma maneira, contribuíram para a realização deste trabalho.

E finalmente,

Graças a Deus por tudo.

Dedico este trabalho a minha querida esposa
Carla pelo seu amor, incentivo e apoio constante.

RESUMO

Este trabalho apresenta um estudo da compressão de imagens usando o padrão JPEG (Joint Photographic Experts Group), cujo comportamento é analisado através de simulações em computador com programas desenvolvidos em linguagem C.

A simulação foi implementada com o objetivo de se estabelecer uma base eficiente para futuras adaptações e otimizações na codificação de imagens paradas, e também visando sua utilização na codificação de imagens em movimento segundo o padrão MPEG (Motion Picture Experts Group).

O sistema foi avaliado através de análises objetivas e subjetivas, principalmente em termos de entropia, relação sinal/ruído e comparação entre os sinais originais e recuperados. Os resultados são de fato promissores para transmissão de imagens paradas e confirmam um bom compromisso entre a qualidade de imagem processada e a compressão de dados.

ABSTRACT

This work is a study on image compression using JPEG standard (Joint Photographics Experts Group), whose performance is evaluated through a computer simulation developed in C Language.

The aim of the simulation was to establish an efficient baseline for future adaptations and optimizations to the encoding of images, and also to enable its utilization in the encoding of moving images according to MPEG standard (Motion Picture Experts Group) .

The system was evaluated through objective and subjective analysis, in terms of entropy, signal to noise ratio and by comparing the original to the recovered signals. The results are promising for the transmission of images taking into account the trade off between the quality of image and the data compression.

SUMÁRIO

| | |
|--|----|
| Introdução..... | 1 |
| | |
| Capítulo 1: Considerações Gerais | |
| 1.1 Formato Digital versus Analógico..... | 3 |
| 1.2 Taxas da Informação de Imagem..... | 3 |
| 1.3 Compressão de Informação vs. Compressão da Largura de Faixa..... | 4 |
| 1.4 Métodos de Compressão..... | 4 |
| 1.5 Aplicações da Compressão..... | 5 |
| 1.6 Taxas de Informação..... | 6 |
| 1.7 Medidas de Fidelidade..... | 7 |
| | |
| Capítulo 2: Técnicas de Compressão de Imagens | |
| 2.1 Introdução..... | 10 |
| 2.2 Modulação por Código de Pulso (PCM)..... | 10 |
| 2.2.1 Codificação Entrópica..... | 12 |
| 2.2.2 Codificação Run-Length (RLC)..... | 16 |
| 2.3 Codificação Preditiva..... | 17 |
| 2.4 Codificação por Transformada..... | 18 |
| 2.5 Codificação Híbrida..... | 20 |
| 2.6 Comentários..... | 21 |
| | |
| Capítulo 3: Codificação por Transformada | |
| 3.1 Introdução..... | 22 |

| | |
|--|----|
| 3.2 Transformada Cosseno de Fourier..... | 22 |
| 3.3 Transformada Cosseno Discreta (DCT)..... | 24 |
| 3.4 Transformada de Karhunen-Loeve (KLT)..... | 28 |
| 3.5 DCT-II..... | 34 |
| 3.6 Transformada Rápida do Cosseno Discreto (Fatorização de Matrizes)..... | 37 |
| 3.7 DCT Bidimensional via DCT Unidimensional..... | 40 |
| 3.8 Comentários..... | 43 |

Capítulo 4: Aspectos Relevantes na Codificação de Imagens segundo o Padrão JPEG

| | |
|---|----|
| 4.1 Introdução..... | 44 |
| 4.2 Classes de Processos de Codificação..... | 45 |
| 4.2.1 Codificação baseada na DCT..... | 45 |
| 4.2.2 Codificação sem Perdas..... | 51 |
| 4.3 Modos de Operação dos Processos de Codificação..... | 52 |
| 4.4 Componentes Múltiplas da Imagem..... | 55 |
| 4.4.1 Formato da Imagem Fonte..... | 55 |
| 4.4.2 Ordem de Codificação..... | 57 |
| 4.5 Tabelas Múltiplas..... | 60 |
| 4.6 Resumo das Características Essenciais dos Processos de Codificação do Padrão JPEG..... | 61 |

Capítulo 5: Simulação e Resultados

| | |
|---------------------------------------|----|
| 5.1 Introdução..... | 62 |
| 5.2 Espaço de Cor..... | 62 |
| 5.3 Modelo JPEG Básico..... | 64 |
| 5.3.1 Transformadas: FDCT e IDCT..... | 65 |

| | | |
|---|---|-----|
| 5.3.2 | Quantização..... | 67 |
| 5.3.3 | Modelo de Codificação..... | 69 |
| 5.3.4 | Codificação Entrópica..... | 71 |
| 5.4 | Simulação e Avaliação do Sistema JPEG Básico..... | 73 |
| 5.4.1 | Plataforma de Trabalho Implementada..... | 74 |
| 5.4.2 | Sinais de Testes..... | 76 |
| 5.4.3 | Capacidade de Compressão..... | 77 |
| 5.4.4 | Avaliação Objetiva..... | 78 |
| 5.4.5 | Avaliação Subjetiva..... | 80 |
| 5.4.6 | Ferramentas Utilizadas..... | 104 |
| Capítulo 6: Conclusões e Sugestões..... | | 106 |
| Referências Bibliográficas..... | | 108 |
| Anexo..... | | 111 |
| Apêndice..... | | 126 |

LISTA DE FIGURAS

| | | |
|------|--|----|
| 2.1 | Sistema PCM..... | 11 |
| 2.2 | Exemplo de um histograma da luminância de uma imagem..... | 12 |
| 2.3 | Exemplo da construção dos códigos de Huffman binários..... | 14 |
| 2.4 | Exemplo da codificação Run-Length..... | 16 |
| 2.5 | Sistema básico preditivo (DPCM)..... | 17 |
| 2.6 | Sistema básico de codificação por transformada..... | 19 |
| 2.7 | Sistema de codificação híbrida..... | 21 |
| 3.1 | Funções Bases para a DCT-II, $N=16$ | 27 |
| 3.2 | Funções Bases da KLT para $N=16$ e $\rho=0,95$ para um sinal Markov 1..... | 33 |
| 3.3 | Vetores bases para um bloco de dimensões 8×8 | 36 |
| 3.4 | Fluxograma do sinal para a DCT-II, $N=16$ | 39 |
| 3.5 | Implementação da DCT 2D $M \times N$ via DCTs 1D..... | 42 |
| 4.1 | Diagrama simplificado do codificador baseado na DCT..... | 45 |
| 4.2 | Preparação dos coeficientes quantizados para a codificação entrópica..... | 46 |
| 4.3 | Diagrama simplificado do decodificador baseado na DCT..... | 47 |
| 4.4 | Imagem Lena e bloco de imagem de 8×8 pixels..... | 47 |
| 4.5 | Diagrama simplificado de um codificador sem perda..... | 52 |
| 4.6 | Predição baseada nas três amostras vizinhas..... | 52 |
| 4.7 | Apresentação progressiva e sequencial..... | 53 |
| 4.8 | Modos de operação dos processos de codificação baseados na DCT..... | 54 |
| 4.9 | Imagem fonte..... | 56 |
| 4.10 | Ordem dos dados para codificação: sem e com interpolação..... | 58 |
| 4.11 | Ordem da codificação interpolada das componentes com dimensões diferentes..... | 59 |

| | | |
|------|--|----|
| 4.12 | Controle de comutação para as tabelas e as componentes..... | 60 |
| 5.1 | Esquema de conversão do espaço RGB 4:4:4 ao espaço YUV 4:2:2..... | 64 |
| 5.2 | Diagrama simplificado do codec JPEG básico..... | 65 |
| 5.3 | Partição e orientação dos blocos de 8x8 amostras..... | 66 |
| 5.4 | Relação entre os blocos de amostras 8x8 e os coeficientes da DCT..... | 69 |
| 5.5 | Seqüência 'zigzag' dos coeficientes quantizados da DCT..... | 70 |
| 5.6 | Seqüência de processamento da plataforma de simulação..... | 74 |
| 5.7 | Janela da plataforma de codificação JPEG básico..... | 75 |
| 5.8 | Janela da plataforma de decodificação JPEG básico..... | 75 |
| 5.9 | Sinais testes do tipo estatísticos: smpte01, smpte02, smpte08 e smpte 15.... | 76 |
| 5.10 | Sinal teste do tipo determinístico: barras coloridas..... | 77 |
| 5.11 | Imagem smpte01 original e recuperada..... | 82 |
| 5.12 | Imagem smpte02 original e recuperada..... | 83 |
| 5.13 | Imagem smpte08 original e recuperada..... | 84 |
| 5.14 | Imagem smpte15 original e recuperada..... | 85 |
| 5.15 | Imagem barras coloridas original e recuperada..... | 86 |
| 5.15 | Imagem barras coloridas original e recuperada..... | 87 |
| 5.16 | Workspace do KHOROS utilizado para a obtenção dos histogramas..... | 88 |
| 5.17 | Histograma da componente R da imagem smpte01..... | 89 |
| 5.18 | Histograma da componente G da imagem smpte01..... | 90 |
| 5.19 | Histograma da componente B da imagem smpte01..... | 91 |
| 5.20 | Histograma da componente R da imagem smpte02..... | 92 |
| 5.21 | Histograma da componente G da imagem smpte02..... | 93 |
| 5.22 | Histograma da componente B da imagem smpte02..... | 94 |
| 5.23 | Histograma da componente R da imagem smpte08..... | 95 |
| 5.24 | Histograma da componente G da imagem smpte08..... | 96 |
| 5.25 | Histograma da componente B da imagem smpte08..... | 97 |
| 5.26 | Histograma da componente R da imagem smpte15..... | 98 |

| | | |
|------|--|-----|
| 5.27 | Histograma da componente G da imagem smpte15..... | 99 |
| 5.28 | Histograma da componente B da imagem smpte15..... | 100 |
| 5.29 | Histograma da componente R da imagem barras coloridas..... | 101 |
| 5.30 | Histograma da componente G da imagem barras coloridas..... | 102 |
| 5.31 | Histograma da componente B da imagem barras coloridas..... | 103 |

LISTA DE TABELAS

| | | |
|------|--|----|
| 2.1 | Quatro distribuições de probabilidade e suas entropias correspondentes..... | 13 |
| 5.1 | Quantização da luminância..... | 68 |
| 5.2 | Quantização da cromaticidade..... | 68 |
| 5.3 | Categorias das magnitudes da diferença..... | 72 |
| 5.4 | Categorias relacionadas aos coeficientes AC..... | 73 |
| 5.5 | Memória ocupada pelos sinais originais e comprimidos..... | 78 |
| 5.6 | MSE e SNR das componentes da imagem..... | 79 |
| 5.7 | Entropia das componentes da imagem..... | 80 |
| 5.8 | Escala de cinco graus para a medida subjetiva da qualidade ou da degradação | 81 |
| 5.9 | Escala de comparação para medida subjetiva da qualidade do sinal recuperado..... | 81 |
| 5.10 | Resultado da avaliação subjetiva..... | 87 |

INTRODUÇÃO

Os avanços na tecnologia digital após a década passada cresceram consideravelmente em relação aos analógicos. Isso, devido as vantagens oferecidas pelo primeiro, tais como: facilidade de armazenamento e recuperação, confiabilidade, compatibilidade com redes digitais, uso conjunto com computadores, etc.

O principal obstáculo para muitas aplicações é a vasta quantidade de informações requerida para representar diretamente a imagem digital com aceitável padrão de qualidade. Por exemplo, a versão digitalizada de um único quadro em cores com resolução de TV contém informações da ordem de um milhão de bytes, enquanto que uma resolução de filme em 35 mm requer 10 vezes essa quantidade. Assim, freqüentemente, o uso de imagens digitalizadas não é viável devido aos altos custos de armazenamento e transmissão, embora estejam disponíveis muitos dispositivos de captura e de visualização de imagens.

A tecnologia moderna de compressão de imagem oferece uma possível solução. O estado atual das técnicas de compressão permite comprimir imagens típicas na razão de 1/10 a 1/50, sem afetar a qualidade da imagem. Mas somente a tecnologia de compressão não é suficiente. Para aplicações da imagem digital, que envolvem armazenamento e transmissão, é necessário um método padrão eficiente de compressão de imagem, que permita a interoperabilidade entre equipamentos de diferentes fabricantes.

Com esse propósito, há alguns anos o grupo conhecido por JPEG (Joint Photographic Experts Group) está trabalhando no sentido de estabelecer um padrão internacional de compressão para satisfazer as necessidades de quase todas as aplicações de imagens “paradas” de tom contínuo na escala de cinza e em cores. Oficialmente o JPEG trabalha em conjunto com o comitê ISO designado JTC1/SC2/WG10 e opera com a colaboração informal do CCITT SGVIII.

Neste trabalho é apresentado um resumo das técnicas mais eficientes de compressão da informação de imagem, bem como os aspectos mais importantes do sistema JPEG para codificação e decodificação de imagens paradas multinível, tais como requerimentos e formas de implementação de seus processos. Outro objetivo deste trabalho é mostrar a viabilidade da elaboração de uma plataforma de simulação do codec JPEG básico, que sirva como base para futuros estudos e otimizações. Cabe ressaltar que o sistema JPEG básico é a base para a codificação de imagens em movimento segundo o padrão MPEG (Moving Picture Experts Group).

Capítulo 1

Considerações Gerais

1.1 Formato Digital versus Analógico

Os dois formatos mais comumente encontrados no processamento de sinais são: o analógico (sinais que são contínuos no espaço e/ou tempo) e o digital (sinais que são usualmente seqüências binárias) [1], sendo os sinais de entrada e saída dos sistemas de televisão, à camera e ao receptor, inerentemente analógicos.

Enquanto as degradações do sinal analógico são acumulativas e difíceis de distinguir do sinal de vídeo, a habilidade de regenerar o trem de pulso digital faz com que os sinais digitais sejam mais resistentes às deficiências do circuito e erros na transmissão; facilitando também a transmissão, armazenamento e processamento de sinais, além do potencial para permitir privacidade e segurança nas comunicações [2].

1.2 Taxas da Informação de Imagem

As imagens típicas de televisão possuem uma resolução espacial de aproximadamente 512 x 512 pixels por quadro, com uma resolução de intensidade de pixel de 8 bit/pixel e 30 quadros/s; o que nos leva a uma taxa aproximada de 60×10^6 bit/s.

Dependendo das aplicações, as taxas de informação de imagem digital crua variam de 10^5 bit/quadro a 10^8 bit/quadro ou mais. Assim, os grandes necessidades de memória e/ou capacidade de canal para o armazenamento ou transmissão de imagens digitais torna imprescindível a utilização de técnicas de compressão de informação [3].

1.3 Compressão de Informação versus Compressão da Largura de Faixa

O processo de conversão de um sinal de vídeo analógico em um sinal digital, incrementa os requisitos de largura de faixa para transmissão. Por exemplo, um sinal de televisão a 4 MHz amostrado a uma taxa de Nyquist de 8 bits por amostra, requer uma largura de faixa de 32 MHz quando é transmitido usando um esquema de modulação digital, tal como PSK (Phase Shift Keying); o qual requer 1 Hz por 2 bits [4].

Enquanto a informação digitalizada tem suas vantagens em termos de flexibilidade de processamento, acesso fácil e aleatório no armazenamento, maior relação sinal/ruído, possibilidade de transmissão sem erros, etc., o preço a pagar está em termos da largura de faixa. Assim, as técnicas de compressão tentam minimizar esse custo utilizando a flexibilidade de processamento para reduzir as necessidades de largura de faixa do sinal digital.

1.4 Métodos de Compressão

Os métodos de compressão de informação de imagem podem ser classificados basicamente em duas categorias diferentes [4]. Na primeira, chamada de codificação preditiva, explora-se a redundância presente na informação. Redundância é uma característica que está relacionada com fatores como predictibilidade, aleatoriedade, similaridade, etc. existentes na informação. Assim muitos algoritmos tentam representar um arranjo original de dados de imagem por um outro arranjo sem redundância, de tal forma que o arranjo original possa ser recuperado a partir desse último, o qual determina a taxa de informação, o tempo de transmissão, a largura de faixa e a quantidade de volume de memória de armazenamento. Técnicas como a Modulação Delta (MD) e Modulação por Código de Pulso Diferencial (DPCM) encontram-se nessa categoria.

Na segunda categoria, chamada de codificação por transformada, a compressão é alcançada transformando-se uma dada imagem em outro arranjo de dados, tal que uma

grande quantidade de informações é empacotada dentro de um pequeno número de amostras. Existem outros algoritmos de compressão de informação de imagem que são generalizações ou combinações desses dois métodos.

O processo de compressão digital em geral resulta em alguma distorção inevitável devido à conversão analógico-digital e aos processos de quantização envolvidos. Além disso, muitos esquemas de compressão de faixa eliminam a informação irrelevante, ou seja, aquela que transporta uma quantidade relativamente insignificante de informação perceptível para os propósitos do receptor.

1.5 Aplicações da Compressão

A compressão é aplicada principalmente na transmissão e no armazenamento de informação [5]. Para transmissão, as técnicas de compressão estão restritas pelas considerações de tempo real e on-line, que tendem a limitar severamente o tamanho e a complexidade do hardware. Para aplicações de armazenamento, os requisitos do compressor são menos severos, devido a que muitos dos pré-processamentos podem ser feitos off-line. Entretanto, a decodificação ou descompressão deve ser rápida e eficiente para minimizar a recuperação ou tempo de resposta.

As aplicações na transmissão de imagem são em TV comercial, sensoriamento remoto via satélite, radar, sonar, teleconferência, comunicações de computadores, transmissão via fac-símile, etc. O armazenamento de imagem é requerido freqüentemente por documentos educacionais e comerciais, imagens médicas usadas em sistemas de monitoramento de pacientes, etc.

1.6 Taxas de Informação

A taxa de informação de imagem crua não representa necessariamente sua taxa média de informação [4]. Por exemplo, as imagens monocromáticas geralmente são quantizadas a 8 bits, dando a taxa de informação de 8 bits por pixel. Entretanto, a taxa de informação média é dada pela entropia (medida em bits):

$$H(u) = - \sum_{i=1}^L P_i \log_2 P_i, \quad (1.1)$$

onde P_i é a probabilidade de que uma amostra u quantizada assumira o valor r_i de um conjunto de $L=2^n$ valores, onde n é o número de bits por pixel. Este é chamado de entropia de ordem nula já que não considera o fato de que uma dada amostra poderia ter dependência estatística sobre seus vizinhos. Para imagens monocromáticas a entropia de primeira ordem é definida como,

$$H(u_k | u_{k-1}) = - \sum_{i_1=1}^L \sum_{i_2=1}^L P_{i_1, i_2} \log_2 (P_{i_1, i_2} / P_{i_2}), \quad (1.2)$$

onde $P_{i_1, i_2} = \text{Prob} [u_k = r_{i_1}, u_{k-1} = r_{i_2}]$; u_{k-1} é um pixel prévio e u_k e P_{i_1}, P_{i_2} são as probabilidades marginais de u_k e u_{k-1} , respectivamente. As entropias de segunda ordem e ordens superiores podem ser definidas similarmente.

De acordo com o teorema da codificação sem ruído de Shannon [6], é possível codificar, sem distorção, uma fonte de entropia H bit/amostra usando $H+\epsilon$ bit/amostra onde ϵ é uma quantidade arbitrariamente positiva e pequena. A máxima compressão C , definida por

$$C = \frac{\text{Taxa média de bits da informação original crua } (n)}{\text{Taxa média de bits da informação codificada } (H+\epsilon)}, \quad (1.3)$$

é $n/(H + \varepsilon) \approx n/H$. A computação de tal taxa de compressão para imagens é impraticável, se não impossível. Por exemplo, uma imagem digital de $N \times M$ pixels com n bits por pixel é uma das $L=2^{nNM}$ possíveis formas de imagem que poderiam ocorrer. Assim, se P_i , a probabilidade da i -ésima forma de imagem fosse conhecida, poderia se computar a entropia, que é a taxa de informação para n bits por pixel $N \times M$ imagens. Depois, podem-se armazenar todas as L possíveis formas de imagem e codificar a imagem pelo seu endereço usando um método de codificação adequado, o qual requererá aproximadamente H bits por imagem ou H/NM bits por pixel.

A principal dificuldade deste método é que ainda para valores relativamente pequenos de N e M , L pode ser proibitivamente grande; por exemplo para $n=8$ e $N=M=16$, $L=2^{2048} \approx 10^{614}$.

A entropia de uma imagem pode ser estimada por sua entropia condicional. Para um bloco de N pixels u_0, u_1, \dots, u_{N-1} , com n bits por pixel e ordenados numa ordem arbitrária, a entropia condicional de N -ésima ordem é definida como:

$$H_N = - \sum_{u_0} \dots \sum_{u_{N-1}} p(u_0, u_1, \dots, u_{N-1}) \log_2 p(u_0 | u_1, \dots, u_{N-1}), \quad (1.4)$$

onde cada u_i , $i=0, \dots, N-1$, leva 2^n valores, e $p(\dots)$ representam as probabilidades relevantes. Para imagens monocromáticas de 8 bits, geralmente as entropias de ordem zero e segunda ordem estão na faixa de 2 a 6 bits/pixel. Teoricamente, quando $N \rightarrow \infty$, H_N converge a H , a entropia por pixel. Segundo a teoria de Shannon, a taxa de bits de qualquer método de codificação exato nunca pode ser menor que a entropia H .

1.7 Medidas de Fidelidade

As técnicas comumente empregadas para a compressão de imagem produzem uma degradação na imagem recuperada. Uma medida da fidelidade da imagem

recuperada, usada para uma imagem de tamanho $N \times M$ é o erro quadrático médio definido como,

$$e_{ms}^2 = \frac{1}{NM} \sum_{i=1}^N \sum_{j=1}^M E(u_{i,j} - \bar{u}_{i,j})^2. \quad (1.5)$$

onde $\{u_{i,j}\}$ e $\{\bar{u}_{i,j}\}$ são respectivamente os valores das amostras da imagem original e reconstruída de tamanho $N \times M$. Mas, experimentalmente, o erro quadrático médio é frequentemente estimado pelo erro quadrático da amostra de uma imagem, dado por,

$$e_{ms}^2 \approx \frac{1}{NM} \sum_{i=1}^N \sum_{j=1}^M (u_{i,j} - \bar{u}_{i,j})^2. \quad (1.6)$$

Existem duas definições da relação sinal/ruído que são usadas com o erro anterior:

$$\text{SNR} = 10 \log_{10} \frac{X_p^2}{e_{ms}^2} \quad (\text{dB}), \quad (1.7)$$

onde X_p^2 é o valor pico a pico da informação de imagem original, e:

$$\text{SNR}' = 10 \log_{10} \frac{\sigma_u^2}{e_{ms}^2} \quad (\text{dB}), \quad (1.8)$$

onde σ_u^2 é a variância da imagem original.

Embora o SNR' seja mais usado na literatura para o processamento de sinal como uma medida de SNR (desde que está relacionado à potência do sinal e à potência do ruído), e é talvez mais significativo porque dá 0 dB para potências de sinal e ruído iguais, o SNR é mais usado no campo de codificação de imagens.

Freqüentemente, a informação de imagem original crua é dada como amostras discretas quantizadas com um número relativamente grande de níveis de cinza.

Tipicamente, o número de níveis é de 256 (ou 8 bits) tal que o valor pico a pico é 255. Portanto,

$$\text{SNR} = 10 \log_{10} \frac{(255)^2}{e_{\text{ms}}^2} \quad (\text{dB}). \quad (1.9)$$

Na eq. (1.9) o valor do erro e_{ms} pode variar entre -255 e +255.

Capítulo 2

Técnicas de Compressão de Imagens

2.1 Introdução

O objetivo principal da codificação (compressão) é representar uma imagem com o menor número possível de bits, preservando o nível da qualidade requerida por uma aplicação específica. Um método óbvio de compressão de informação de imagem seria reduzir a taxa de amostragem, os números de níveis de quantização e taxa de repetição (número de quadros por segundo para imagens em movimento). Mas, à medida que se reduz a quantidade de informação, ocorrem os fenômenos associados, tais como, sobreposição (aliasing), falso contorno (contouring) e oscilação (flickering). Assim, segundo esses métodos citados, a compressão dirigida para imagens de alto padrão de qualidade é relativamente pequena quando comparada com outras técnicas mais avançadas e já disponíveis as quais serão brevemente revisadas a seguir.

2.2 Modulação por Código de Pulso (PCM)

O processo PCM, denominado conversor analógico/digital, representa o tempo (ou espaço) e a amplitude da informação visual na forma discreta. Como mostra a fig. 2.1, o PCM consiste em amostrar uma forma de onda (usualmente a uma taxa maior que a de Nyquist) e quantizar cada amostra usando 2^K níveis. Cada nível é representado por uma palavra binária de K bits.

Embora não seja mostrado na fig. 2.1, um pré-filtro é usado antes da amostragem para evitar as distorções de sobreposição quando a taxa de amostragem é próxima à taxa de Nyquist.

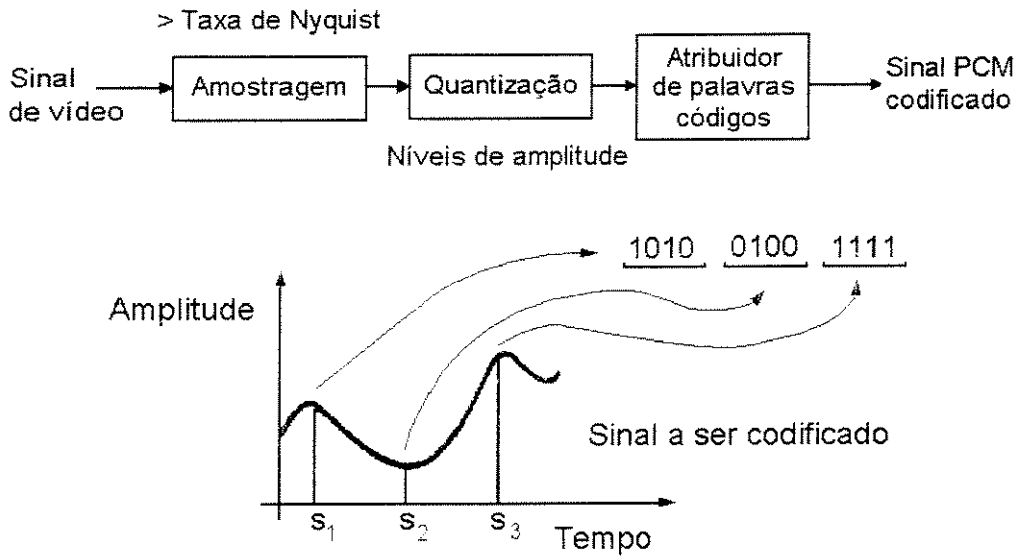


Fig. 2.1: Sistema PCM.

Comumente, 8 bits são suficientes para imagens monocromáticas de rádio difusão ou videoconferência. Entretanto, as imagens em sistemas médicos ou sinais de vídeo em cores podem requerer de 10 a 12 bits por pixel. O número de bits de quantização necessário para visualização de imagens pode ser reduzido a 4 bits por pixel usando compensadores, quantização por contraste ou técnicas “dithering”. As técnicas de “half-tone” reduzem a saída do quantizador a 1 bit por pixel, mas usualmente a taxa de amostragem de entrada deve ser incrementada por um fator de 2 a 16. A compressão alcançada por essas técnicas é geralmente menor que 2:1 [4].

Em termos da distorção quadrática média, a taxa mínima alcançável pelo PCM é dada pela fórmula da taxa de distorção:

$$R_{\text{PCM}} = \frac{1}{2} \log_2 \frac{\sigma_u^2}{\sigma_q^2}, \quad \sigma_q^2 < \sigma_u^2 \quad (2.1)$$

onde σ_u^2 é a variância da entrada do quantizador e σ_q^2 é a distorção quadrática média do quantizador. Dessa forma, a taxa de bits na linha é igual ao produto da frequência de amostragem pelo número de bits por amostra. Em geral utiliza-se uma frequência de amostragem fixa (constante). Logo os métodos de compressão trabalham no sentido de

diminuir o número de bits por amostra. Usualmente essa redução do número de bits por amostra é feita na média ao longo do tempo.

Embora o PCM seja bastante simples quando comparado com as outras técnicas de codificação, apresenta-se ineficiente no sentido de que não elimina nenhuma redundância presente no sinal de quadro.

2.2.1 Codificação Entrópica

Quando uma imagem monocromática é mapeada, amostrada (para produzir MN pixels) e quantizada a K bits, o PCM requererá MNK bits para representar a imagem. Supondo que a ocorrência dos valores quantizados da luminância não seja probabilisticamente igual, como é usual (por exemplo ver o histograma da fig.2.2) é possível reduzir o número de bits requeridos para representar a imagem se, em vez de atribuir palavras de K bits a cada um dos 2^K níveis de luminância, atribuem-se palavras de vários comprimentos.

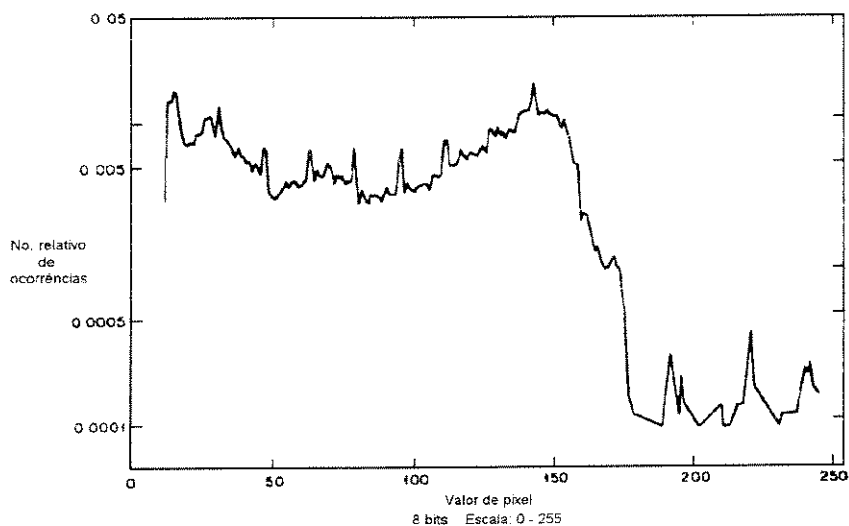


Fig. 2.2: Exemplo de um histograma da luminância de uma imagem.

Codificação de Huffman

A codificação de Huffman é um método de codificação com comprimento variável de palavra código, que minimiza a taxa de bits média da eq. (2.2) e onde suas palavras de comprimento médio \bar{L}_1 encontram-se no intervalo [8]:

$$H \leq \bar{L}_1 \leq H+1 \quad \text{bit/pixel.} \quad (2.4)$$

A construção de um código de Huffman envolve o uso de uma árvore. Como exemplo mostra-se a seguinte figura.

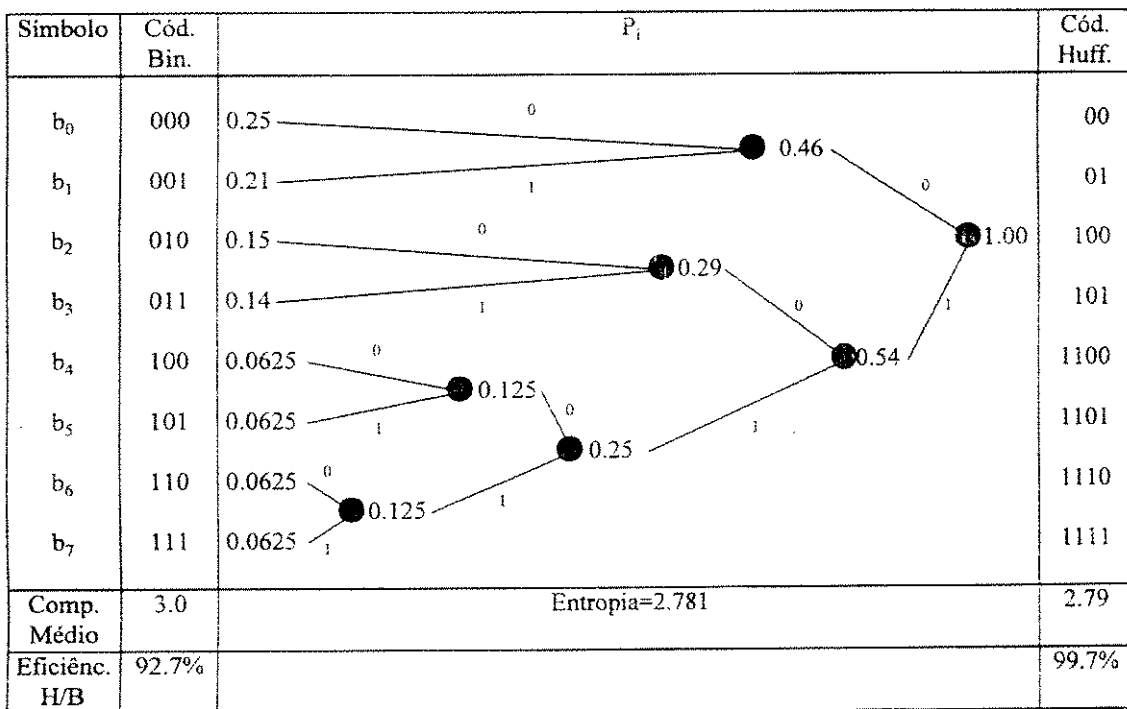


Fig. 2.3: Exemplo da construção dos códigos de Huffman binários.

Nesse exemplo, consideram-se os elementos de uma imagem quantizados com 3 bits ou seja com 8 níveis, aos quais estão associados suas probabilidades. Os níveis $\{b_i\}$ são primeiramente ordenados pelas probabilidades em forma decrescente, e enquanto exista mais de um nó em cada seção, os dois nós com menor probabilidade são combinados para formar um novo nó e cuja probabilidade é a soma das probabilidades dos dois nós. Arbitrariamente, atribui-se 1 e 0 para cada par de ramos de um nó. Finalmente para a construção do código, lê-se seqüencialmente do topo até a raiz do nó, determinando-se o código. Nesse exemplo, a taxa de bits média \bar{L} é 2.71 bits por pixel e a entropia é 2.68 bits por pixel.

O algoritmo apresentado proporciona o chamado dicionário de códigos de Huffman para qualquer conjunto de probabilidades dado. A codificação e decodificação são realizadas simplesmente procurando-se os valores numa tabela.

Embora o código de Huffman minimize o comprimento de palavra médio, na prática outros códigos são também usados. Por exemplo, os códigos com um número pequeno de comprimento de palavra são freqüentemente mais fáceis de se implementar do que os códigos de Huffman. Também os códigos com comprimento de palavra muito compridos são usualmente evitados. Embora os códigos de comprimento de palavra fixo requeiram mais bits, possuem a vantagem de que no caso de uma transmissão com erro, o começo da palavra código seguinte é fácil de se encontrar. Entretanto, os códigos com comprimento variável, em particular o código de Huffman perdem a sincronização no caso de transmissão com erro. Isto é, alguns códigos podem ser decodificados incorretamente antes que a decodificação correta seja restabelecida. Também, durante o período da perda da sincronização, os pixels poderiam ser eliminados ou pixels extras adicionados. O histograma da imagen de televisão é aproximadamente uniforme, embora as estatísticas sejam altamente estacionárias. Conseqüentemente a codificação entrópica não é muito prática para informações de imagens cruas. Entretanto, é bastante utilizada nos algoritmos de codificação preditiva e por transformada, e também para a codificação de informação binária, tal como gráficos e imagens de facsimile [4].

2.2.2 Codificação Run-Length (RLC)

Na codificação run-length uma seqüência de pixels u_1, u_2, \dots, u_N ao longo de uma linha é mapeada para uma seqüência de pares de inteiros $(r_1, l_1), (r_2, l_2), \dots, (r_K, l_K)$, onde r_K denota o nível de cinza e l_K denota o comprimento de corrida (corrente), ou seja, o número de pixels justapostos, tendo o mesmo nível de cinza.

A fig. 2.4 ilustra um exemplo do método de codificação “run-length”. Nesse exemplo, uma forma de se realizar a implementação do bloco de imagem original de 8 x 4 amostras, cada amostra com 8 bits por pixel, que requer um total de $8 \times 4 \times 8 = 256$ bits na codificação PCM, é reordenada na forma de par inteiros (r, l) , assumindo 8 bits para os níveis de cinza e 3 bits para a corrida, resultando num total de $14 \times 11 = 154$ bits. Assim, a razão de compressão alcançada nesse caso é de 1.66:1 ou 4.81 bits por pixel.

Seqüência de dados da imagem original

| | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 255 | 250 | 250 | 250 | 250 | 236 | 236 | 236 |
| 230 | 240 | 240 | 240 | 240 | 240 | 240 | 236 |
| 224 | 224 | 240 | 240 | 240 | 240 | 224 | 210 |
| 224 | 224 | 224 | 240 | 240 | 240 | 210 | 190 |

Mapeamento (r, l)

| | | | | | | | | | |
|-----|---|-----|---|-----|---|-----|---|-----|---|
| 255 | 1 | 250 | 4 | 236 | 3 | 230 | 1 | 240 | 6 |
| 236 | 1 | 224 | 2 | 240 | 4 | 224 | 1 | 210 | 1 |
| 224 | 3 | 240 | 3 | 210 | 1 | 190 | 1 | | |

Fig 2.4: Exemplo da codificação Run-Length.

Esse método de codificação é sempre útil quando as imagens apresentam corridas de zeros compridas. Como exemplos têm-se documentos impressos, gráficos, mapas atmosféricos, imagens após codificação por transformada e outras mais. A eficiência da codificação “run-length” pode ser melhorada aplicando-se em seguida a codificação com comprimento variável.

2.3 Codificação Preditiva

No sistema PCM, as amostras que entram no quantizador são tratadas independentemente, de tal forma que nenhuma avaliação da redundância estatística do sinal é feita. Entretanto, a filosofia das técnicas preditivas é remover a redundância entre pixels sucessivos e codificar somente a informação nova.

O sistema básico da codificação preditiva [9] é conformado por três elementos básicos (ver fig. 2.5): preditor, quantizador e atribuidor de código.

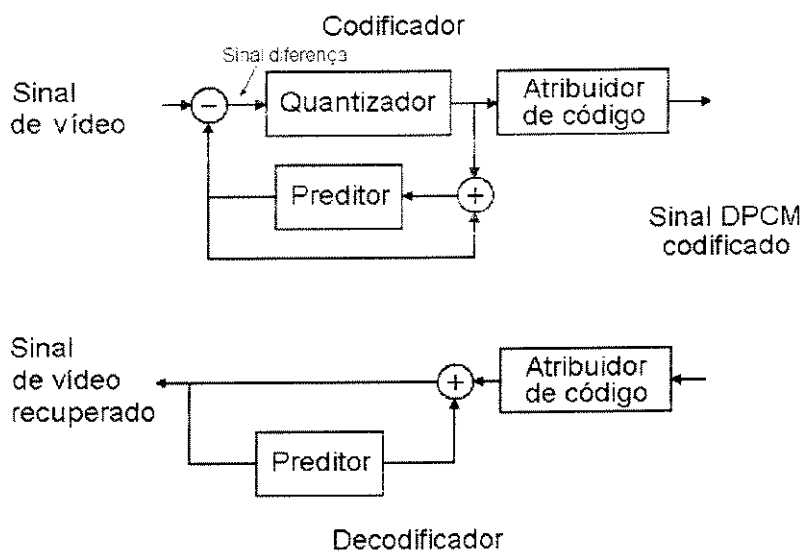


Fig. 2.5: Sistema básico preditivo (DPCM).

Nesse sistema uma predição é feita a partir da informação prévia codificada, que tenha sido transmitida. O sinal diferença (ou erro) resultante da subtração do valor de predição com o valor atual da amostra, é logo quantizado num conjunto de “L” níveis de amplitude discretos. Esses níveis são representados como palavras binárias de comprimento fixo ou variável e entregues ao codificador de canal para serem transmitidos.

A taxa mínima alcançada pelo DPCM é dada pela fórmula da taxa de distorção [10]:

$$R_{\text{DPCM}} = \frac{1}{2} \log_2 \frac{\sigma_e^2}{\sigma_q^2} \quad \text{bit/pixel,} \quad (2.5)$$

onde σ_q^2 e σ_e^2 são os valores quadráticos médios do erro de quantização e o erro de predição, respectivamente.

Dependendo do número de níveis do quantizador, faz-se distinção entre Modulação Delta (MD) que tem apenas L=2 níveis e a Modulação Diferencial por Código de Pulso (DPCM), onde L tem mais de dois níveis [11]. A DM não tem sido muito utilizada na codificação de imagens, porque para se conseguir uma imagem de qualidade aceitável, a amostragem deve ser feita a uma frequência maior do que a de Nyquist.

2.4 Codificação por Transformada

Na codificação por transformada [2], uma imagem é transformada para um domínio mais conveniente e significativamente diferente do domínio da intensidade da imagem, sendo depois codificados os coeficientes transformados de forma eficiente.

Essa codificação também explora a observação de que para imagens típicas uma quantidade grande de energia está concentrada numa fração pequena dos coeficientes transformados. Esta propriedade é chamada de compactação de energia. Com essa

propriedade é possível codificar somente uma fração dos coeficientes transformados sem afetar seriamente a qualidade subjetiva da imagem, ou seja a degradação pode ser controlada e ser imperceptível às pessoas.

Outra propriedade desejável de uma transformada é a redução de correlação entre os coeficientes transformados, chamada propriedade de redução de correlação. Com a escolha apropriada das funções base, a correlação entre os coeficientes pode ser reduzida.

Um esquema geral para a codificação por transformada é mostrado na fig. 2.6. Nessa figura cada bloco de n pixels é transformado por uma matriz unitária T para um bloco de coeficientes, os quais devem ser eficientemente codificados para alcançar uma compressão significativa, por exemplo de 4:1.

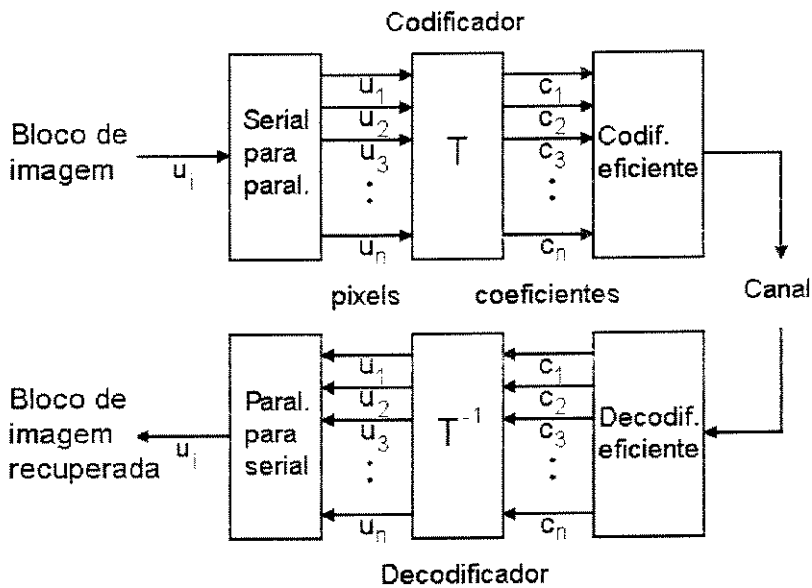


Fig. 2.6: Sistema básico de codificação por transformada.

Uma das transformadas clássicas usadas foi a transformada de Fourier em codificação de imagem, mas hoje tem sido substituída por outras transformadas que tem a vantagem da alta eficiência na codificação com manipulação baseada só em números reais.

Esse método pode ser mais eficiente e menos sensível aos erros, mas envolve complexidade alta e requer mais cálculos do que a codificação preditiva.

2.5 Codificação Híbrida

Além das dificuldades da codificação por transformadas mencionadas anteriormente, tem-se que para fontes de imagem com estatísticas estacionárias, a codificação por transformada com blocos de tamanhos grandes remove essencialmente toda a redundância estatística presente no sinal fonte. Mas na prática, as fontes de imagem não tem estatísticas estacionárias, assim é desejável que os tamanhos de blocos sejam menores para permitir que as técnicas adaptativas possam acomodar as mudanças estatísticas nas áreas locais. Mas utilizando blocos de tamanhos menores se produzirá sempre uma considerável redundância de bloco para bloco após a codificação por transformada.

Assim, a codificação híbrida, cujo termo refere-se à combinação de técnicas de codificação por transformada e preditiva (ver fig. 2.7), é utilizada com o objetivo de se conseguir o melhor compromisso entre a taxa de compressão e qualidade de imagem, procurando-se reduzir os requisitos na quantidade de memória de armazenamento e da complexidade de processamento numérico.

Geralmente, uma imagem bidimensional é primeiro transformada (transformada unitária) em uma de suas dimensões para se obter uma seqüência unidimensional; depois cada seqüência de informação é codificada independentemente por uma técnica preditiva unidimensional, tal como DPCM. Esta técnica combina as vantagens de simplicidade de hardware do DPCM e a eficiência robusta da codificação por transformada.

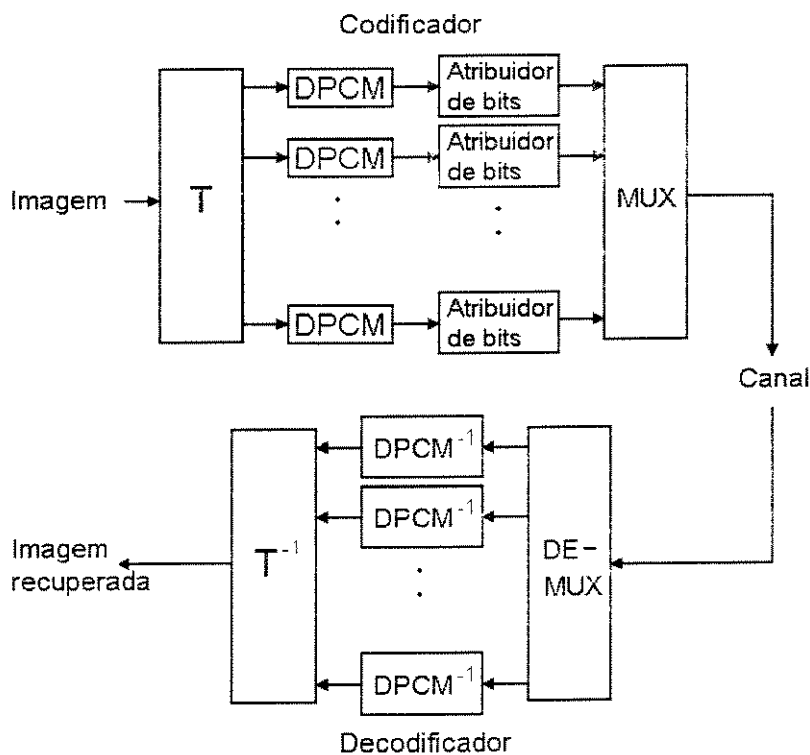


Fig. 2.7: Sistema de codificação híbrida.

2.6 Comentários

Em forma geral, pretendeu-se apresentar algumas das técnicas de compressão de taxa de bits dentro do contexto voltado à transmissão digital de informação de vídeo, e que são utilizadas em forma conveniente pelo o padrão JPEG.

No presente trabalho a ênfase é dada à codificação por transformada, já que é a principal responsável da compressão de imagens segundo o padrão JPEG. As outras técnicas mencionadas neste capítulo são também igualmente importantes já que realizam a codificação eficiente dos coeficientes transformados.

Atualmente existem muitos esquemas sendo pesquisados para atingir o objetivo de redução da taxa de bits de transmissão (sub-bandas, quantização vetorial, wavelets e outros).

Capítulo 3

Codificação por Transformada

3.1 Introdução

As transformadas, em particular as transformadas integrais, são usadas principalmente para a redução da complexidade dos problemas matemáticos. Mas quando as transformadas são aplicadas no processamento digital de sinais, são amplamente usadas para compressão de imagens. Exemplos bem conhecidos são as bases das transformadas de: Fourier, Cosseno, Hadamard e Karhunen-Loeve.

Estudos mostram que a base da transformada Karhunen-Loeve alcança a maior representação de compactação de energia (assumindo que a informação é altamente correlacionada), entretanto a base da transformada Cosseno é muito próxima a essa, com a vantagem de que não é dependente da imagem.

Na discussão da Transformada Cosseno Discreta (DCT), considera-se primeiramente a Transformada Cosseno de Fourier (FCT). Tenta-se tratar a DCT como aproximações da FCT discretizada, o que seria errado, devido a que na realidade trata-se com amostras, medições e instantes de tempo. A continuidade é somente uma idealização para permitir o uso de ferramentas matemáticas apropriadas.

3.2 Transformada Cosseno de Fourier (FCT)

Dada uma função $x(t)$ para $-\infty < t < \infty$, sua transformada de Fourier é dada por [12],

$$X(\omega) \equiv F[x(t)] = \left(\frac{1}{2\pi}\right)^{1/2} \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt, \quad (3.1.a)$$

sujeita às condições de existência usuais para a integral. Tem-se $j=\sqrt{-1}$ e $\omega=2\pi f$ é a frequência radial, sendo f é a frequência em Hertz. A função $x(t)$ pode ser recuperada pela transformada inversa de Fourier, ou seja,

$$x(t) \equiv F^{-1}[X(\omega)] = \left(\frac{1}{2\pi}\right)^{1/2} \int_{-\infty}^{\infty} X(\omega)e^{j\omega t} d\omega. \quad (3.1.b)$$

Se $x(t)$ é definida somente para $t \geq 0$, constrói-se a função $y(t)$ dada por:

$$y(t) = \begin{cases} x(t) & t \geq 0 \\ x(-t) & t \leq 0 \end{cases}$$

Logo,

$$\begin{aligned} F[y(t)] &= \left(\frac{1}{2\pi}\right)^{1/2} \left\{ \int_0^{\infty} x(t)e^{-j\omega t} dt + \int_{-\infty}^0 x(-t)e^{-j\omega t} dt \right\} \\ &= \left(\frac{1}{2\pi}\right)^{1/2} \int_0^{\infty} x(t)[e^{-j\omega t} + e^{j\omega t}] dt \\ &= \left(\frac{2}{\pi}\right)^{1/2} \int_0^{\infty} x(t) \cos(\omega t) dt. \end{aligned} \quad (3.2)$$

Agora pode-se definir esta como sendo a Transformada Cosseno de Fourier (FCT) de $x(t)$ ou seja,

$$X_c(w) \equiv F_c[x(t)] = \left(\frac{2}{\pi}\right)^{1/2} \int_0^{\infty} x(t) \cos(wt) dt. \quad (3.3)$$

Note-se que $X_c(w)$ é uma função par em w . Aplicando-se a inversa de Fourier à eq. (3.3), obtém-se,

$$y(t) = x(t) \equiv F^{-1}[X_c(w)] = \left(\frac{2}{\pi}\right)^{1/2} \int_0^{\infty} X_c(w) \cos(wt) dw \quad (t \geq 0). \quad (3.4)$$

As eqs. (3.3) e (3.4) definem o par (Direta e Inversa) da FCT.

3.3 Transformada Cosseno Discreta (DCT)

Da eq. (3.3) nota-se que a Transformada de Cosseno de Fourier tem uma base dada por,

$$K_c(w, t) = \cos(wt). \quad (3.5)$$

Seja $w_m = 2\pi m \delta f$ e $t_n = n \delta t$ a frequência angular e o tempo amostrado, onde δf e δt representam respectivamente a unidade de intervalos entre amostras para a frequência e o tempo; m e n são inteiros. Agora a eq. (3.5) pode ser escrita na forma,

$$K_c(w_m, t_n) = K_c(2\pi m \delta f, n \delta t) = \cos(2\pi m n \delta f \delta t) = K_c(m, n). \quad (3.6)$$

Seja $\delta f \delta t = 1/2N$, onde N é um inteiro. Tem-se,

$$K_c(m, n) = \cos\left(\frac{\pi m n}{N}\right). \quad (3.7)$$

Esta equação representa a base discreta da transformada cosseno.

Considerando a base (3.7) como elementos de uma matriz transformada $[M]$ de $(N+1) \times (N+1)$, tem-se:

$$[M]_{mn} = \cos\left(\frac{\pi mn}{N}\right) \quad m, n=0, 1, \dots, N \quad (3.8)$$

e aplicando-a a um vetor coluna $x=[x(0), x(1), \dots, x(N)]^T$, obtém-se o vetor $X=[X(0), X(1), \dots, X(N)]^T$, i.e.,

$$X = [M] x,$$

onde

$$X(m) = \sum_{n=0}^N \cos\left(\frac{\pi mn}{N}\right) x(n) \quad m=0, 1, \dots, N. \quad (3.9)$$

Diz-se que o vetor x sofreu uma transformação discreta. A transformada discreta do cosseno na eq. (3.9) foi relatada pela primeira vez por Kitajima em 1980 [13] e foi chamada Transformada Cosseno Simétrica (SCT).

Em geral, desde que a inversão é parte do processamento da transformada, as matrizes não singulares são usualmente preferidas. Entretanto, existem transformadas que são singulares ou não inversíveis. Um exemplo desse tipo é a Transformada Rapid (RT). Na prática, se a matriz não singular é real e ortogonal, sua inversa como sua transposta são facilmente obtidas, tais matrizes denominam-se matrizes de transformada unitária.

As definições para as quatro transformadas discretas do cosseno, classificadas por Wang [14], são apresentadas:

1. DCT-I:

$$\left[C_{N+1}^I \right]_{mn} = \left(\frac{2}{N} \right)^{1/2} \left[K_m K_n \cos \left(\frac{mn\pi}{N} \right) \right] \quad m,n = 0, 1, \dots, N;$$

2. DCT-II

$$\left[C_N^{II} \right]_{mn} = \left(\frac{2}{N} \right)^{1/2} \left[K_m \cos \left(\frac{m(n+1/2)\pi}{N} \right) \right] \quad m,n = 0, 1, \dots, N-1;$$

3. DCT-III

$$\left[C_N^{III} \right]_{mn} = \left(\frac{2}{N} \right)^{1/2} \left[K_n \cos \left(\frac{(m+1/2)n\pi}{N} \right) \right] \quad m,n = 0, 1, \dots, N-1;$$

4. DCT-IV

$$\left[C_N^{IV} \right]_{mn} = \left(\frac{2}{N} \right)^{1/2} \left[\cos \left(\frac{(m+1/2)(n+1/2)\pi}{N} \right) \right] \quad m,n = 0, 1, \dots, N-1;$$

onde,

$$K_j = \begin{cases} 1 & j \neq 0 \text{ ou } N \\ 1/\sqrt{2} & j = 0 \text{ ou } N \end{cases}$$

Nota-se que a DCT-II é a primeira Transformada Cosseno Discreta reportada por Ahmed, Natarajan e Rao [15]. A DCT-III é obviamente a transposta da DCT-II e a DCT-IV é a versão deslocada de DCT-I. As funções bases da DCT-II para $N=16$ são mostradas na fig. 3.1.

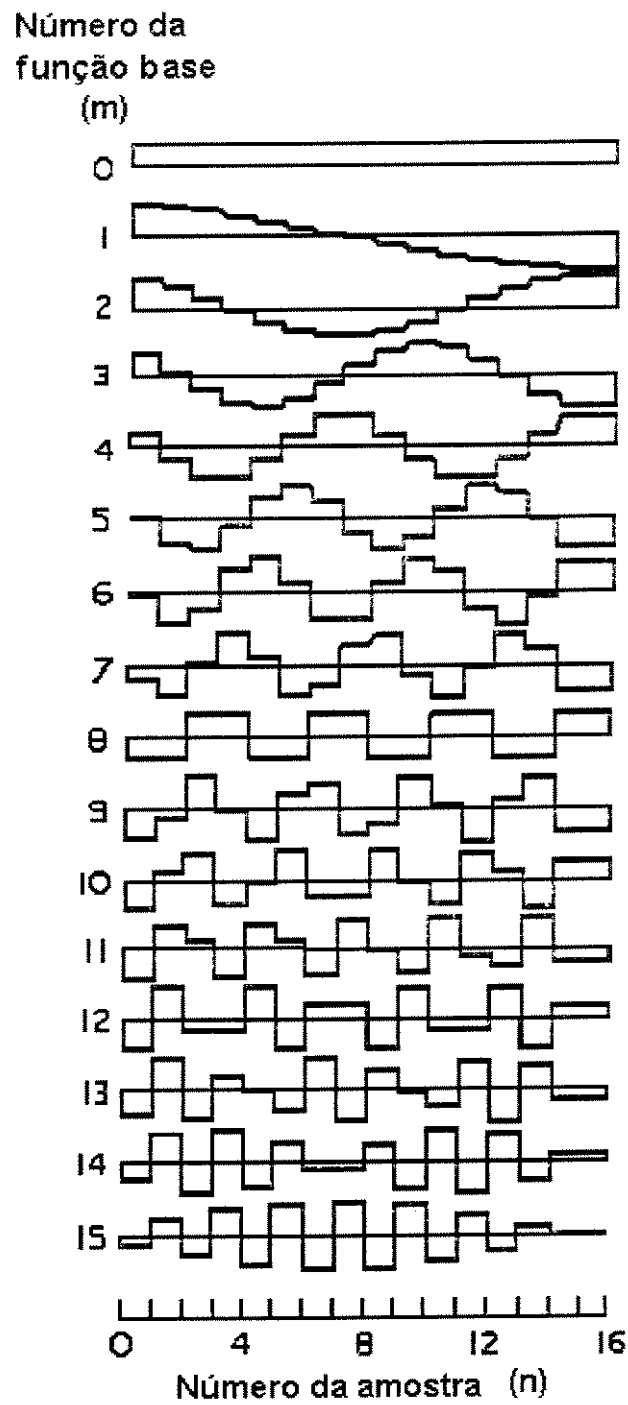


Fig. 3.1: Funções bases para a DCT-II, $N=16$.

3.4 Transformada de Karhunen-Loeve (KLT)

É útil considerar a transformada de Karhunen-Loeve (KLT) de um ponto de vista intuitivo. Supondo que um sinal na forma de uma senóide pura tem que ser transmitido através de algum tipo de meio, o sinal pode ser transmitido como uma forma de onda amostrada, com cada ponto de informação (amostra) a ser entregue em forma seqüencial. O número de pontos transmitidos depende de como se deseja reconstruir a forma de onda. Intuitivamente, com mais pontos transmitidos, melhor é a forma de onda reconstruída. Entretanto, sabe-se que tudo o que é necessário para reconstruir uma sinusóide determinística é sua magnitude, fase, frequência, tempo de início e o fato de que é uma forma de onda conhecida, ou seja uma senóide. Isso implica que cinco peças de informação é tudo o que se necessita para se reconstruir exatamente o sinal.

Do ponto de vista teórico, os valores amostrados são altamente correlacionados e o conteúdo de informação do sinal é baixo. Por outro lado, as cinco peças de informação são completamente descorrelacionadas e têm exatamente a mesma quantidade do conteúdo de informação como o número total de valores amostrados a serem transmitidos. A tentativa de pegar os N pontos amostrados e transformá-los em cinco peças descorrelacionadas de informação, dá origem às transformadas. A KLT é justamente uma transformada ótima para sinais do tipo Markov-1.

Outra maneira de formular a KLT [16] é considerar a melhor representação de uma função aleatória. Considerem-se N pontos amostrados correspondentes a um vetor aleatório \mathbf{x} de média zero,

$$\mathbf{x} = \{x_0, x_1, \dots, x_{N-1}\}^T.$$

Se $\{\phi_i\}$ é um conjunto de vetores linearmente independentes desse espaço vetorial N -dimensional, então \mathbf{x} pode ser expandido em termos de ϕ_i ,

$$\mathbf{x} = \sum_{i=0}^{N-1} X_i \phi_i, \quad (3.10)$$

onde X_i são os coeficientes de expansão dados por,

$$X_i = \frac{\langle x, \phi_i \rangle}{\langle \phi_i, \phi_i \rangle} \quad i = 0, 1, \dots, N-1, \quad (3.11)$$

onde o vetor x pode ser representado por N números x_i ou como o número de X_i no espaço definido pelas funções bases ϕ_i .

Na eq. (3.11), supondo que somente os primeiros D coeficientes ($D < N$) são significativamente diferentes de zero, o vetor x pode ser bem representado pelos D coeficientes no espaço $\{\phi_i\}$. Isto é, em vez de N números, somente D de tais números são necessários para representar x . Na prática, isso significa compressão de informação e redução da largura de faixa.

Assim, dado um vetor aleatório $x = \{x(0), x(1), \dots, x(N-1)\}^T$, deseja-se encontrar um conjunto de vetores bases ϕ_i , $i = 0, \dots, N-1$, tal que o erro da representação truncada seja minimizado pelo MSE (Minimum Square Error).

A representação truncada de x é dada por,

$$\tilde{x} = \sum_{i=0}^{D-1} X_i \phi_i. \quad (3.12)$$

O MSE no truncamento é dado por,

$$\varepsilon = E[(x - \tilde{x})^2] = E\left[\left\langle \sum_{i=D}^{N-1} X_i \phi_i, \sum_{i=D}^{N-1} X_i \phi_i \right\rangle\right], \quad (3.13)$$

onde E é o operador de esperança matemática e $\langle \rangle$ é o produto interno dos argumentos.

Assumindo que as funções bases são ortonormais, tal que,

$$\langle \phi_i, \phi_k \rangle = \delta_{ik} \quad (3.14)$$

e que x é real, obtém-se,

$$\varepsilon = E \left[\sum_{i=D}^{N-1} |X_i|^2 \right] = E \left[\sum_{i=D}^{N-1} |\langle x, \phi_i \rangle|^2 \right]. \quad (3.15)$$

A equação anterior pode reduzir-se a:

$$\varepsilon = E \left[\sum_{i=D}^{N-1} \phi_i^T x x^T \phi_i \right] = \sum_{i=D}^{N-1} \phi_i^T E [x x^T] \phi_i, \quad (3.16)$$

onde T denota a transposição. Define-se a matriz de autocovariância do vetor aleatório x como,

$$[A] = E [x x^T] \quad (3.17)$$

e minimiza-se a quantidade ε da eq. (3.16) pela escolha apropriada das funções bases, sujeitas à condição de normalização da eq. (3.14), isto é.,

$$\left(\frac{\partial}{\partial \phi_i} \right) \{ \varepsilon - \mu_i \langle \phi_i, \phi_i \rangle \} = 0.$$

Obtém-se, assim

$$([A] - \mu_i [I_N]) \phi_i = 0 \quad i = 0, 1, \dots, N-1, \quad (3.18)$$

onde μ_i é o multiplicador de Lagrange e $[I_N]$ é a matriz identidade $N \times N$. Note-se também que $[A]$ é simétrica e positiva semi-definida.

Assim, o conjunto de vetores bases deverá diagonalizar a matriz de autocovariância $[A]$. Logo,

$$[\phi]^{-1} [A] [\phi] = \text{diag} [\mu_0, \mu_1, \dots, \mu_{N-1}]. \quad (3.19)$$

Baseando-se no que foi mostrado anteriormente, o MSE devido ao truncamento é dado por,

$$\varepsilon = \sum_{i=D}^{N-1} \mu_i. \quad (3.20)$$

O conjunto de vetores bases $\{\phi_i\}$ formam as bases para a expansão da KLT e a matriz $[\phi]$ é a matriz da KLT. A KLT é uma transformada ótima devido a que:

- Descorrelaciona completamente o sinal no domínio da transformada.
- Minimiza o MSE na redução da largura de faixa ou na compressão de informação.
- Concentra a máxima energia média no menor número de coeficientes da transformada.
- Minimiza a representação entrópica total da seqüência.

Estas são conseqüências diretas do problema de diagonalização da eq. (3.18). Está claro que as funções bases são dependentes da matriz de autocovariança $[A]$ e por isso não podem ser predeterminadas. Além disso, dada uma matriz de autocovariança, a solução da eq. (3.18) é usualmente bastante complicada. Existem somente poucos casos no qual as soluções analíticas são disponíveis. Tal é o caso quando os sinais estatísticos produzem uma matriz de autocovariança da forma,

$$[A]_{ik} = \rho^{|i-k|} \quad i, k = 0, 1, \dots, N-1, \quad (3.21)$$

para $0 < \rho < 1$, onde ρ é o coeficiente de correlação adjacente. Este sinal é denominado sinal estacionário Markov-1.

Davenport e Root [17] reportaram tal exemplo no domínio contínuo como uma solução de uma equação integral. No caso discreto, Ray e Driver [18] proporcionaram a solução como:

$$\begin{aligned} \phi]_{mn} &= \phi_m(n) \\ &= \left[\frac{2}{N + \mu_m} \right]^{1/2} \text{sen} \left\{ w_m \left[(n+1) - \frac{(N+1)}{2} \right] + (m+1) \frac{\pi}{2} \right\} \end{aligned} \quad (3.22)$$

$$m, n = 0, 1, \dots, N-1,$$

onde

$$\mu_m = \frac{1 - \rho^2}{1 - 2 \cos(w_m) + \rho^2}, \quad (3.23)$$

são os autovalores e w_m são as raízes reais e positivas da equação transcendental:

$$\tan(Nw) = - \frac{(1 - \rho^2) \text{sen}(w)}{[\cos(w) - 2\rho + \rho^2 \cos(w)]}, \quad (3.24)$$

$[\phi]_{mn}$ denota o elemento-mn da matriz $[\phi]$ e $\phi_m(n)$ é a n-ésima componente do m-ésimo autovetor, onde $m, n = 0, 1, \dots, N-1$.

Na Fig. 3.2 mostram-se as funções bases na expansão K-L para um sinal estacionário Markov-1 com $\rho=0.95$ e $N=16$. Nota-se a natureza senoidal das funções bases.

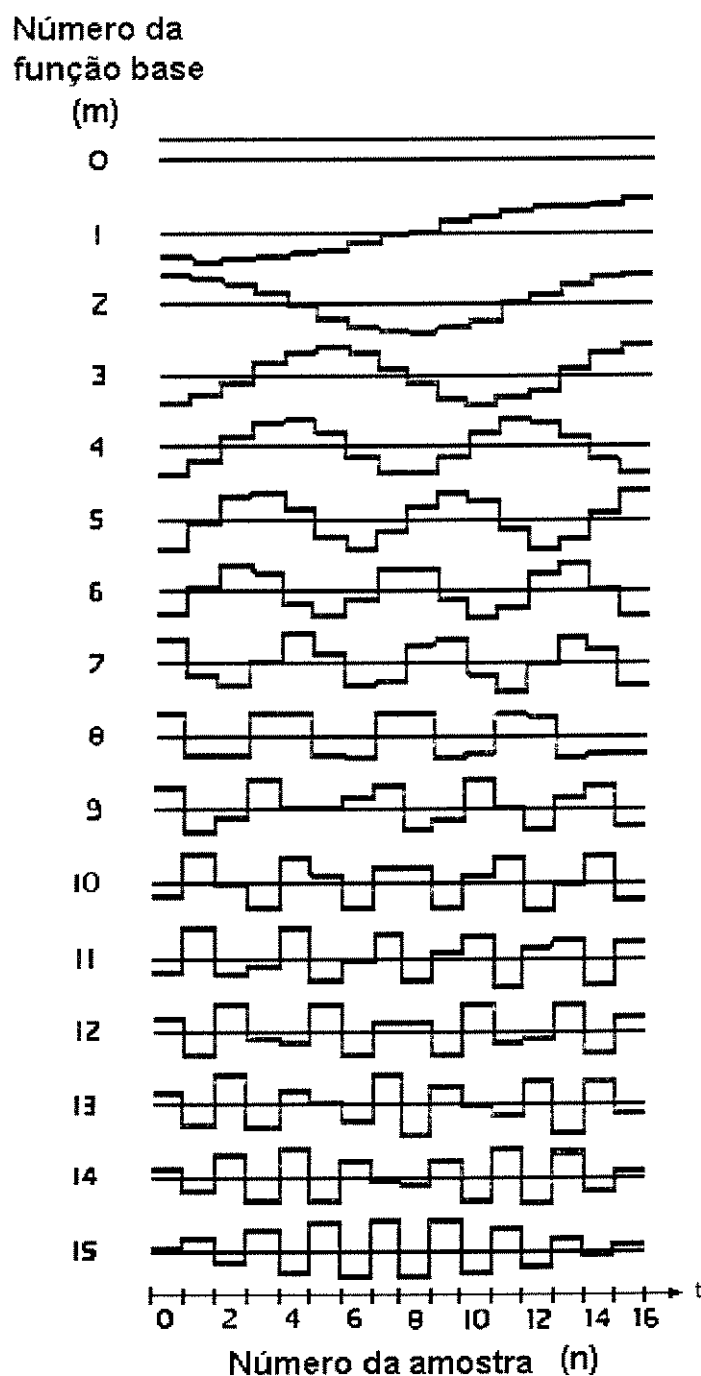


Fig. 3.2: Funções Bases da KLT para $N=16$ e $\rho=0.95$ para um sinal Markov-1.

Na prática, a implementação da KLT envolve a estimativa da matriz de autocovariância da seqüência de informação, sua diagonalização e a construção dos vetores bases. A incapacidade de predeterminar os vetores bases no domínio da transformada tem feito da KLT uma ferramenta ideal, mas impraticável. Assim, procuram-se vetores bases predeterminados que sejam aproximações boas da KLT, por meio da análise da diagonalização das matrizes que sejam assintoticamente equivalentes à eq. (3.21). Embora a KLT não proporcione soluções fáceis ao problema de decorrelacionamento, serve para comparar e julgar as outras transformadas.

3.5 DCT-II

A derivação da DCT-II para sinais Markov-1 [16], baseados na equivalência assintótica à KLT, considera os valores limites para o coeficiente de correlação. Das equações (3.22), (3.23) e (3.24), quando $\rho \rightarrow 1$, tem-se a equação transcendental,

$$\tan(Nw)=0. \quad (3.25)$$

Assim define-se:

$$w_k = \frac{k\pi}{N} \quad k = 0, 1, \dots, N-1. \quad (3.26)$$

Os autovalores μ_m desaparecem quando w_m não são nulos. Para μ_m , quando $m=0$, a expressão aproxima-se do infinito. Entretanto, da expressão:

$$\sum_{m=0}^{N-1} [A]_{mn} = \sum_{m=0}^{N-1} \mu_m, \quad (3.27)$$

e como os elementos da diagonal de $[A]$ num sinal Markov-1 são todos 1, tem-se que $\mu_0=N$. Combinando esta e substituindo em (3.22), obtém-se,

$$[\phi]_{n0} = 1/\sqrt{N} \quad (3.28)$$

e

$$\begin{aligned} [\phi]_{nm} &= \sqrt{\frac{2}{N}} \operatorname{sen} \left[m(n+1/2) \frac{\pi}{2} + \frac{\pi}{2} \right] \\ &= \sqrt{\frac{2}{N}} \cos \left[m(n+1/2) \frac{\pi}{N} \right], \quad m \neq 0. \end{aligned} \quad (3.29)$$

Simplificando, obtém-se a base DCT-II:

$$[\phi]_{nm} = \sqrt{\frac{2}{N}} K_m \cos \left[m(n+1/2) \frac{\pi}{N} \right] \quad m, n = 0, 1, \dots, N-1, \quad (3.30)$$

com

$$K_m = \begin{cases} 1 & m = 0 \\ 1/\sqrt{2} & m \neq 0 \end{cases}$$

Quando $\rho \rightarrow 0$, a eq. (3.24) é reduzida a,

$$\tan(Nw) = -\tan w,$$

proporcionando,

$$w_n = \frac{(n+1)\pi}{(N+1)} \quad n = 0, 1, \dots$$

e todo $\mu_n = 1$. Pode-se mostrar que, neste caso as funções bases correspondentes são uma das transformadas do seno discretas [19]. Entretanto, deve-se ter em mente que como ρ tende a zero, a matriz de covariância se aproxima da $[I_N]$, que já é diagonal. Assim, pode-

se observar que a DCT-II é assintoticamente equivalente à KLT para sinais Markov-1 quando $\rho \rightarrow 1$.

Os coeficientes da transformada DCT bi-dimensional definem os vetores bases, que são dados por,

$$\phi_{m_1, m_2}(n_1, n_2) = \frac{2}{\sqrt{N_1 N_2}} K_1 K_2 \cos \left[(2n_1 + 1) \frac{\pi m_1}{2N_1} \right] \cos \left[(2n_2 + 1) \frac{\pi m_2}{2N_2} \right], \quad (3.31)$$

$$m_1, m_2, n_1, n_2 = 0, 1, \dots, N-1.$$

A fig. 3.3 mostra uma visualização dos vetores bases para $N \times N = 8 \times 8$.

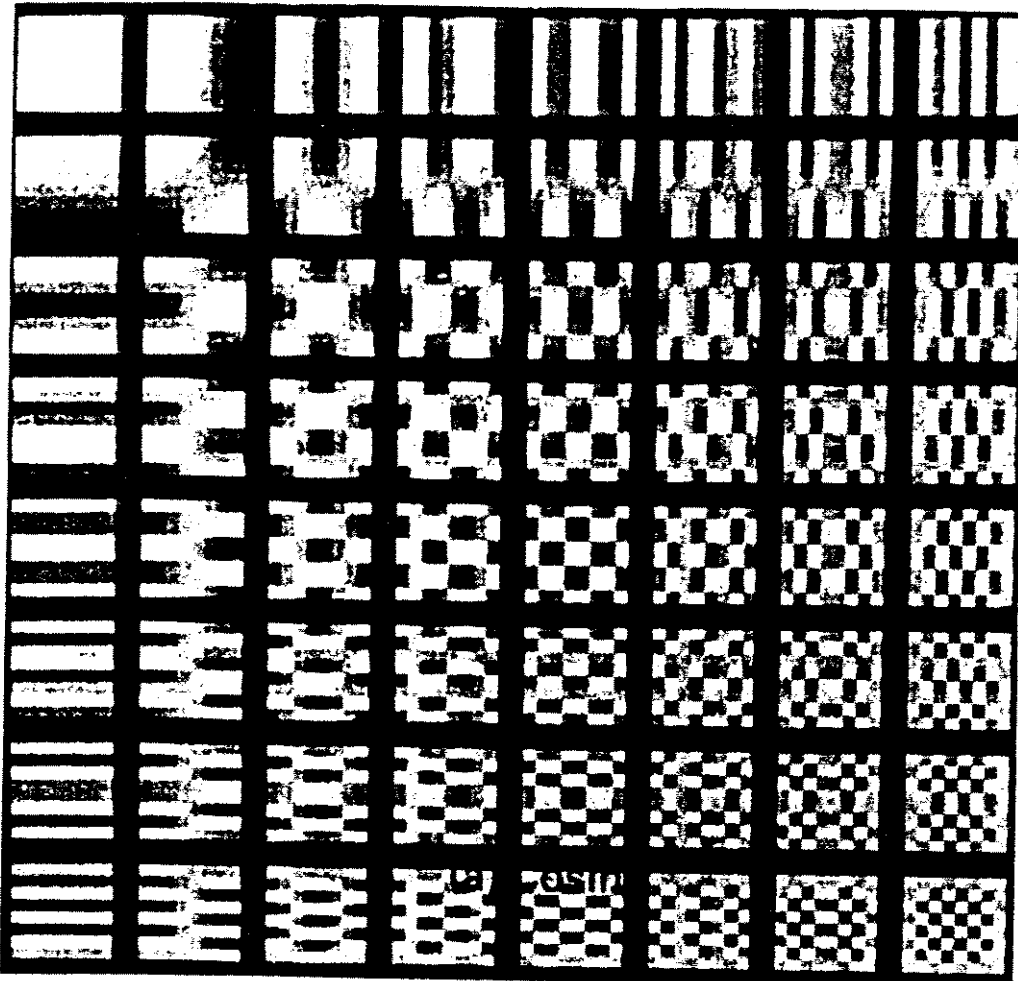


Fig. 3.3: Vetores bases para um bloco de dimensões 8 x 8.

3.6 Transformada Rápida do Cosseno Discreta (Fatoração de Matrizes).

A Transformada Discreta do Cosseno de um vetor de informação $[x]$ de $N \times 1$ pode ser expressa na forma de matriz como,

$$[X] = \left(\frac{2}{N}\right)^{1/2} [A_N]x, \quad (3.32)$$

onde $[A_N]$ é uma matriz $N \times N$ e é igual a $[C_N^m] = K_m \cos[m(n+1/2)\pi/N]$, onde $m, n=0, 1, \dots, N-1$.

O algoritmo para a computação rápida a ser apresentado foi desenvolvido por Chen, Smith e Fralick [20] e baseia-se na decomposição matricial da matriz $[A_N]$. Mostram-se $[A_2]$, $[A_4]$ e uma decomposição para $[A_4]$.

$$[A_2] = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ C_4^1 & C_4^3 \end{bmatrix} \quad \text{e} \quad [A_4] = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ C_8^1 & C_8^3 & C_8^5 & C_8^7 \\ C_8^2 & C_8^6 & C_8^6 & C_8^2 \\ C_8^3 & C_8^7 & C_8^1 & C_8^5 \end{bmatrix},$$

onde $C_k^i = \cos[i\pi/k]$ é usado para simplificar as notações. A matriz $[A_4]$ pode ser decomposta como,

$$[A_4] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ C_4^1 & C_4^3 & C_4^3 & C_4^1 \\ C_8^3 & -C_8^1 & C_8^1 & -C_8^3 \\ C_8^1 & C_8^3 & -C_8^3 & -C_8^1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 & 0 \\ C_4^1 & C_4^3 & 0 & 0 \\ 0 & 0 & -C_8^1 & C_8^3 \\ 0 & 0 & C_8^3 & C_8^1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \quad (3.33)$$

A eq. (3.33) pode ser expressa na forma mais compacta como,

$$[A_4] = [\bar{P}_4] \begin{bmatrix} [A_2] & 0 \\ 0 & [\bar{R}_2] \end{bmatrix} [B_4], \quad (3.34)$$

onde $[\bar{P}_4]$ é uma matriz de permutação, $[B_4]$ é uma matriz borboleta e $[\bar{R}_2]$ é o bloco principal 2x2 na segunda matriz fator. O objetivo é que $[A_4]$ seja reduzida em termos de $[A_2]$. Em geral para N, uma potência de 2, tem-se,

$$[A_N] = [\bar{P}_N] \begin{bmatrix} [A_{N/2}] & 0 \\ 0 & [\bar{R}_{N/2}] \end{bmatrix} [B_N], \quad (3.35)$$

onde $[\bar{P}_N]$ permuta as filas pares em ordem crescente da metade superior e as filas ímpares em ordem decrescente da metade inferior. A matriz borboleta $[B_N]$ pode ser expressa em termos da matriz identidade $[I_{N/2}]$ e da matriz identidade oposta $[\tilde{I}_{N/2}]$, como,

$$[B_N] = \begin{bmatrix} [I_{N/2}] & [\tilde{I}_{N/2}] \\ [\tilde{I}_{N/2}] & -[I_{N/2}] \end{bmatrix}. \quad (3.36)$$

A matriz $[\bar{R}_N]$ é obtida invertendo-se as ordens das linhas e colunas da matriz $[R_N]$, cujos ik elementos são dados por,

$$[R_N]_{ik} = \cos \left[\frac{(2i+1)(2k+1)\pi}{4} \right] \quad i, k = 0, 1, \dots, N-1 \quad (3.37)$$

Observe que em (3.35) a fatoração é somente parcialmente recursiva, já que a matriz $[R_N]$ não pode ser recursivamente fatorada. Entretanto, existem regularidades em sua fatoração, e encontra-se que este pode ser decomposto em cinco tipos de matrizes fatores, os quais não têm mais que dois elementos não nulos em cada fila. Mais detalhes dos fatores matrizes para $[R_N]$ podem ser encontrados nos trabalhos de Wang [21] e Chen, Smith e Fralick [12].

Na fig. 3.4 é mostrado o diagrama do sinal para o caso da DCT-II para $N=16$, onde $X_c(m) = X^{(2)}(m)$.

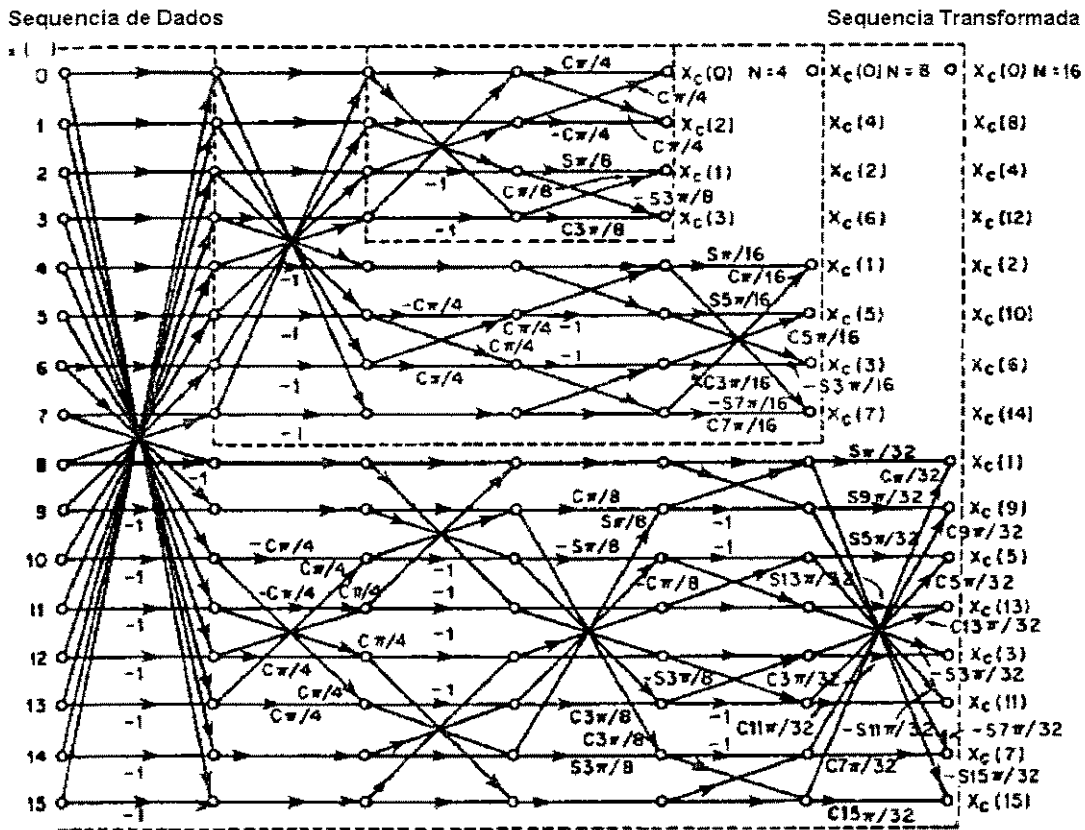


Fig. 3.4: Fluxograma do sinal para a DCT-II, $N=16$.

Este algoritmo requer,

$$\frac{3N}{2}(\log_2 N - 1) + 2 \quad \text{adições reais}$$

$$N \log_2 N - \frac{3N}{2} + 4 \quad \text{multiplicações reais.}$$

3.7 DCT Bidimensional via DCT Unidimensional.

Seja $[g]$ uma matriz ($M \times M$) representando uma imagem digital em preto e branco (por simplicidade), onde o elemento g_{mn} poderia ser interpretado como o nível de cinza ou a intensidade de pixel na posição (m,n) . $[G]$, a transformada ortogonal bidimensional da matriz $[g]$, e sua inversa são respetivamente definidas como,

$$[G] = [T_M][g][T_N]^T \quad (3.38.a)$$

e

$$[g] = [T_M]^T [G] [T_N], \quad (3.38.b)$$

onde $[T_M]$ e $[T_N]$ são respectivamente as matrizes transformadas reais ($M \times M$) e ($N \times N$). A forma da eq. (3.38) assume que a transformada bidimensional pode ser implementada por uma série de transformadas unidimensionais. Esta propriedade é válida para qualquer transformada separável, tal como, a DCT, WHT, DFT, ST, HT, etc.

Supondo que as linhas (ou colunas) da matriz $[g]$ são simplesmente concatenadas para criar um vetor coluna x de MN componentes, o vetor transformado unidimensional X é dado por,

$$X = [T_{MN}]x, \quad (3.39)$$



onde $[T_{MN}]$ é de dimensão $(MN \times MN)$. Observe que as eqs. (3.38) e (3.39) são muito diferentes e que a eq. (3.39) envolve $(MN \times MN)$ elementos na transformação, entretanto a eq. (3.38) envolve $(M \times M + N \times N)$ elementos. Assim, a simples concatenação para reduzir a transformada bidimensional a uma unidimensional requer considerações adicionais.

Para o caso da DCT, seja $[g]$ uma matriz $(M \times M)$ representando informação bidimensional e $[G]$ a DCT-II bidimensional de $[g]$. Logo uv elementos de $[G]$ são dados por,

$$G_{uv} = \frac{2c(u)c(v)}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} g_{mn} \cos\left[\frac{(2m+1)u\pi}{2M}\right] \cos\left[\frac{(2n+1)v\pi}{2N}\right], \quad (3.40.a)$$

onde $u = 0, \dots, M-1$, $v = 0, \dots, N-1$ e

$$c(k) = \begin{cases} 1/\sqrt{2} & k = 0 \\ 1 & k \neq 0 \end{cases}$$

Similarmente, o mn elemento de $[g]$ é dado por a IDCT-II bidimensional de $[G]$, definido como,

$$g_{mn} = \frac{2}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} c(u)c(v)G_{uv} \cos\left[\frac{(2m+1)u\pi}{2M}\right] \cos\left[\frac{(2n+1)v\pi}{2N}\right], \quad (3.40.b)$$

onde $m = 0, \dots, M-1$ e $n = 0, \dots, N-1$. A propriedade de separabilidade da DCT bidimensional (3.38) pode ser ilustrada como:

$$G_{uv} = \sqrt{\frac{2}{M}}c(u) \sum_{m=0}^{M-1} \left\{ \sqrt{\frac{2}{N}}c(v) \sum_{n=0}^{N-1} g_{mn} \cos\left[\frac{(2n+1)v\pi}{2N}\right] \right\} \cdot \cos\left[\frac{(2m+1)u\pi}{2M}\right], \quad (3.40.c)$$

onde $u = 0, \dots, M-1$ e $v = 0, \dots, N-1$.

A somatória interna é uma DCT-II unidimensional de N pontos das filas de [g], a somatória externa representa a DCT-II unidimensional de M pontos das colunas da matriz semitransformada. Isso implica que a DCT bidimensional (MN) pode ser implementada por DCTs de MxN pontos ao longo das filas de [g], seguidas por DCT de NM pontos ao longo da matriz obtida depois da transformação da fila. A ordem em que se transforma a fila e a coluna é indiferente, pode-se tomar a ordem inversa nos dois conjuntos da DCT unidimensional, como mostra a seguinte equação,

$$G_{uv} = \sqrt{\frac{2}{N}}c(v) \sum_{n=0}^{N-1} \left\{ \sqrt{\frac{2}{M}}c(u) \sum_{m=0}^{M-1} g_{mn} \cos \left[\frac{(2m+1)u\pi}{2M} \right] \right\} \cdot \cos \left[\frac{(2n+1)v\pi}{2N} \right]. \quad (3.40.d)$$

A descrição feita de reduzir a transformada bidimensional a uma unidimensional pode ser resumida na fig. 3.5.

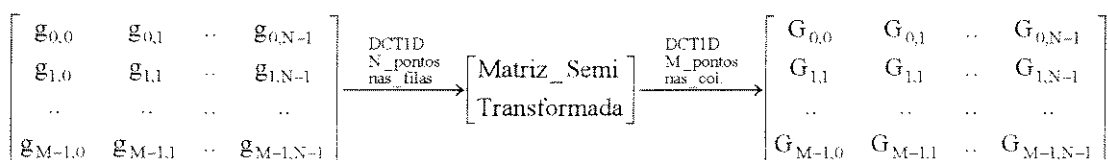


Fig. 3.5: Implementação da DCT 2D MxN via DCTs 1D.

Observando a eq. (3.40.a) e (3.40.b) mostra-se que a propriedade de separabilidade é igualmente válida para a IDCT 2D. De fato, essa propriedade pode ser estendida a dimensões maiores que dois.

3.8 Comentários

Resumindo, as vantagens da DCT são: transformada ortogonal e separável, próxima a compactação ótima de energia e tem muitos algoritmos rápidos para seu cálculo com complexidade computacional baixa. Além disso, para uma transformada ortogonal discreta, como a DCT, se a transformada é separável, também o é a inversa. Se uma transformada 1D direta tem um algoritmo rápido, também a transformada multidimensional direta terá. Para uma transformada ortogonal, essa propriedade é também válida para a transformação inversa. Além disso, se uma arquitetura particular é utilizada para implementar uma transformada 1D em hardware, o mesmo conceito pode ser estendido para a transformada multidimensional [22].

Capítulo 4

Aspectos Relevantes na Codificação de Imagens segundo o Padrão JPEG

4.1 Introdução

Em junho de 1987, o JPEG dirigiu o processo de seleção para estabelecer um método padrão de codificação de imagens paradas de tons contínuos, que fosse capaz de proporcionar imagens de boa qualidade, taxas de compressão de pelo menos 16:1, tratável complexidade computacional e permitir os diferentes modos de operação (seqüencial, progressivo, hierárquico e sem perda) [23].

Dos doze métodos propostos [24], três foram selecionados: ADCT (Adaptative Discrete Cosine Transform), ABAC (Adaptative Binary Arithmetic Differential Pulse Code Modulation) e PCS (Progressive Coding Scheme).

Em janeiro de 1988, revelou-se que o proposto ADCT baseado na DCT 8x8 foi o mais eficiente, sendo selecionado para representar o algoritmo do sistema JPEG. Até então, o método baseado na DCT foi parcialmente definido para alguns modos de operação.

De 1988 a 1990, o grupo JPEG conseguiu dar maior impulso ao trabalho de melhorar o modelo, definindo, documentando, testando, avaliando e aumentando os detalhes necessários para uma genuína interoperabilidade [26]. Os esforços consideráveis desenvolvidos são apresentados neste capítulo.

4.2 Classes de Processos de Codificação

Tipicamente o padrão JPEG suporta dois tipos de processos de codificação: com perda e sem perda.

Os processos de codificação com perda são aqueles baseados na DCT, que permitem uma compressão considerável enquanto produzem uma imagem reconstruída com uma elevada fidelidade visual relativa à imagem fonte do codificador.

Os processos de codificação sem perda não se baseiam na DCT, e foram propostos para satisfazer certas necessidades onde se requer que a imagem reconstruída seja idêntica à imagem fonte do codificador.

4.2.1 Codificação baseada na DCT

A fig. 4.1 mostra os procedimentos fundamentais de todos os processos de codificação que se baseiam na DCT [16]. Por simplicidade, ilustra-se o caso especial de uma imagem com uma componente.

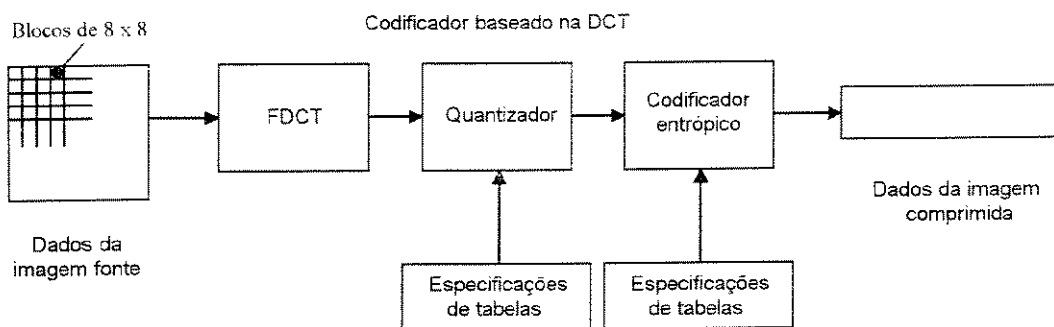


Fig.4.1: Diagrama simplificado do codificador baseado na DCT.

No processo de codificação, as amostras da componente de entrada são agrupadas em blocos de 8x8 amostras, e cada bloco é transformado por meio da DCT direta num conjunto de 64 valores denominados coeficientes DCT, concentrando a

energia de informação dentro dos primeiros e poucos coeficientes. A um desses se denomina coeficiente DC (constante) e aos outros 63 coeficientes AC (variáveis). Cada um dos 64 coeficientes são quantizados e preparados para serem codificados entropicamente, como mostra a fig.4.2. O coeficiente DC previamente quantizado é utilizado para prever o coeficiente atual DC quantizado. A diferença entre eles (DPCM) será codificada entropicamente. Entretanto, os 63 coeficientes AC são ordenados numa seqüência “zigzag” unidimensional, para colocar primeiro os coeficientes mais significativos (de mais baixa freqüência), seguidos pelos menos significativos (de mais alta freqüência) e favorecer a formação de filas ou corridas (correntes) de zeros. Uma codificação vantajosa neste ordenamento é a codificação Run-Length. Em seguida, estes passam por um procedimento de codificação entrópica (Huffman ou Aritmética), que comprime ainda mais os dados.

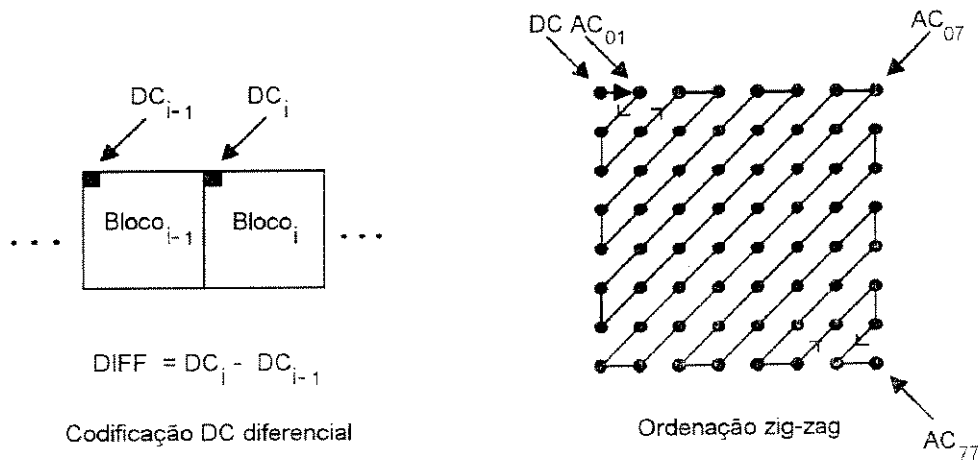


Fig. 4.2: Preparação dos coeficientes quantizados para a codificação entrópica.

A fig. 4.3 mostra os procedimentos principais de todos os processos de decodificação baseados na DCT.

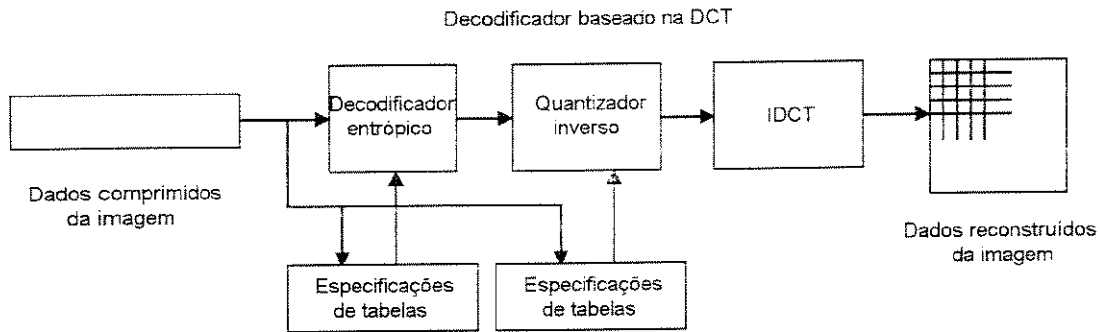


Fig. 4.3: Diagrama simplificado do decodificador baseado na DCT.

Cada passo mostrado realiza basicamente o inverso do processo equivalente correspondente no codificador. O decodificador de entropia decodifica a sequência “zigzag” dos coeficientes quantizados da DCT. Depois da quantização inversa, os coeficientes da DCT são transformados num bloco de 8x8 amostras por meio de uma DCT inversa.

Exemplo 4.1: Na fig. 4.4 se apresenta a imagem Lena, da qual se extraiu um bloco de tamanho 8x8 pixels.

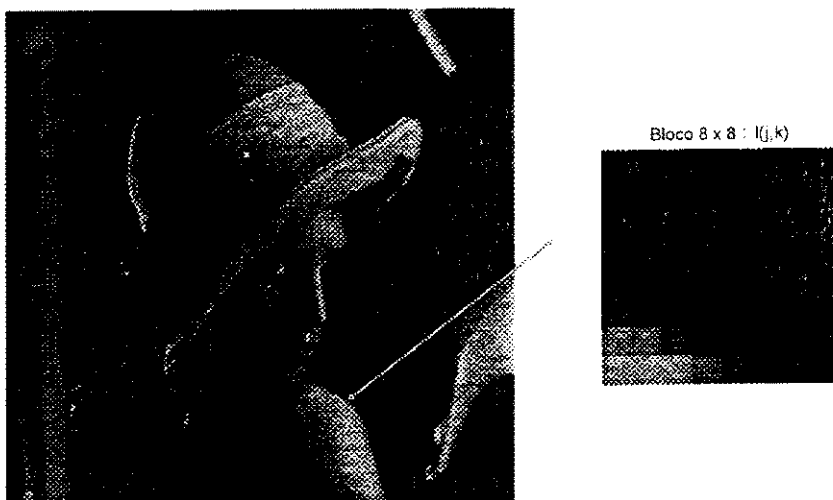


Fig.4.4: Imagem Lena e bloco de imagem de 8 x 8 pixels.

- Os valores de intensidade do bloco extraído $I(j,k)$ são:

$I(j,k)$

| | | | | | | | |
|-----|-----|-----|-----|-----|-----|----|----|
| 63 | 57 | 59 | 59 | 65 | 62 | 58 | 52 |
| 57 | 58 | 58 | 61 | 61 | 59 | 62 | 60 |
| 57 | 55 | 56 | 55 | 53 | 57 | 63 | 54 |
| 66 | 55 | 53 | 53 | 46 | 52 | 57 | 48 |
| 123 | 100 | 68 | 52 | 49 | 52 | 54 | 49 |
| 179 | 152 | 124 | 90 | 60 | 54 | 50 | 46 |
| 208 | 200 | 180 | 150 | 109 | 75 | 54 | 45 |
| 209 | 211 | 209 | 195 | 169 | 139 | 97 | 58 |

- A DCT do bloco 8×8 $I(j,k)$ é dada por :

$$DCT_{uv} = \frac{K_u K_v}{4} \sum_{i=0}^7 \sum_{j=0}^7 I_{ij} \cos[(2j+1)u\pi/16] \cos[(2k+1)v\pi/16],$$

onde $K_x = 1/\sqrt{2}$, $x=0$ e $K_x=1$ para $x \neq 0$, $x=u$ ou v .

Os coeficientes DCT_{uv} correspondentes ao bloco selecionado são dados por:

DCT_{uv}

| | | | | | | | |
|----------|----------|---------|---------|--------|--------|--------|-------|
| 680.125 | 181.917 | 9.434 | 12.613 | -4.875 | 9.881 | -1.067 | 0.288 |
| -254.881 | -183.904 | 10.359 | 1.206 | 3.917 | -0.812 | -1.351 | 2.959 |
| 147.960 | 44.073 | -57.515 | -5.025 | -1.288 | -4.334 | 0.722 | 1.473 |
| -36.184 | 38.061 | 46.199 | 1.953 | 5.658 | -3.132 | 2.596 | 0.429 |
| 4.625 | -29.118 | -16.644 | 16.205 | -1.875 | 0.182 | -0.006 | 0.684 |
| -12.972 | 13.897 | -4.004 | -10.253 | 2.241 | 2.489 | 1.370 | 0.335 |
| 5.415 | -4.650 | 1.222 | 5.695 | -4.743 | 3.463 | 2.765 | 1.073 |
| 0.977 | 5.838 | 4.140 | 3.038 | 0.555 | -4.620 | -2.370 | 0.962 |

- Os coeficientes transformados $DCT(u,v)$ são quantizados como sendo :

$$QDCT(u,v) = \text{inteiro mais próximo } [DCT(u,v)/Q(u,v)],$$

onde $Q(u,v)$ é a matriz de quantização por “default”, cujos valores estão entre 1 a 255.

A matriz $Q(u,v)$ é definida por :

$$Q(u,v) = (\text{FATOR_ESCALA}) \times (\text{default_}Q(u,v))$$

O FATOR_ESCALA igual a 1 implica na utilização da matriz de quantização default_ $Q(u,v)$. Um FATOR_ESCALA maior produz uma quantização grosseira, introduzindo erros na reconstrução do bloco 8x8. Não entanto, produz uma maior compressão. Por outro lado, um FATOR_ESCALA menor produz uma quantização mais fina, produzindo menor erro na reconstrução e uma menor compressão.

As entradas na matriz default_ $Q(u,v)$ são determinadas levando-se em conta a importância de seus correspondentes coeficientes no domínio da DCT.

Uma matriz default_ $Q(u,v)$ típica é dada por:

default_ $Q(u,v)$

| | | | | | | | |
|----|----|----|----|-----|-----|-----|-----|
| 16 | 11 | 10 | 16 | 24 | 40 | 51 | 61 |
| 12 | 12 | 14 | 19 | 26 | 58 | 60 | 55 |
| 14 | 13 | 16 | 24 | 40 | 57 | 69 | 56 |
| 14 | 17 | 22 | 29 | 51 | 87 | 80 | 62 |
| 18 | 22 | 37 | 56 | 68 | 109 | 103 | 77 |
| 24 | 35 | 55 | 64 | 81 | 104 | 113 | 92 |
| 49 | 64 | 78 | 87 | 103 | 121 | 120 | 101 |
| 72 | 92 | 95 | 98 | 112 | 100 | 103 | 99 |

Para um FATOR_ESCALA igual a 1, tem-se que $QDCT(u,v)$ vale :

QDCT(u,v)

| | | | | | | | |
|-----|-----|----|---|---|---|---|---|
| 42 | 16 | 0 | 0 | 0 | 0 | 0 | 0 |
| -21 | -15 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 3 | -3 | 0 | 0 | 0 | 0 | 0 |
| -2 | 2 | 2 | 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 |

Note-se que o processo de quantização produz muitos valores nulos nas altas frequências.

- A matriz DCT quantizada QDCT(u,v) é reordenada numa sequência “zig-zag” uni-dimensional:

42 , 16 , -21 , 10 , -15 , 0 , 0 , 0 , 3 , -2 , 0 , 2 , -3 , 0 , 0 , 0 , 0 , 0 , 2 , -1,

- O termo DC (= 42) é codificado diferencialmente com respeito ao termo DC do bloco prévio. Esse valor é então codificado diferencialmente e depois é codificado entropicamente.
- A varredura “zig-zag” é feita para produzir uma sequência adequada para a codificação “run-length”. Assim, esta é representada como um par de valores dado por :

(corrida, nível)

| (CORRIDA,NIVEL) | |
|-----------------|-----|
| 0 | 16 |
| 0 | -21 |
| 0 | 10 |
| 0 | -15 |
| 3 | 3 |
| 0 | -2 |
| 1 | 2 |
| 0 | -3 |
| 5 | 2 |
| 0 | -1 |

- A saída é remapeada como: (0,16), (0,-21), (0,10), (0,-15), (3,3), (0,-2), (1,2), (0,-3), (5,2), (0,-1) e (EOB) e finalmente essa seqüência de pares é codificada entropicamente. Sendo que EOB (End of the Block) é o fim do bloco, ou seja, quando os coeficientes restantes do bloco são todos de valor nulo.

4.2.2 Codificação sem Perdas

A fig. 4.5 mostra os procedimentos principais dos processos de codificação sem perda, onde um preditor combina os valores reconstruídos de até três amostras vizinhas nas posições a, b e c para elaborar uma predição da amostra x, como indica a fig. 4.6. Depois se subtrai essa predição do valor real da amostra da posição x, e a diferença codifica-se entropicamente sem perdas.

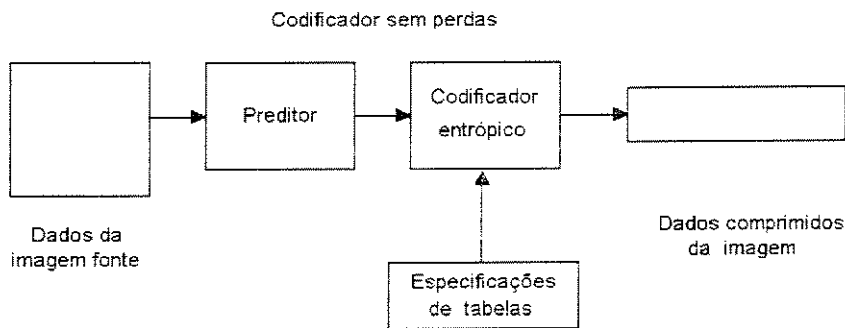


Fig. 4.5: Diagrama simplificado de um codificador sem perda.

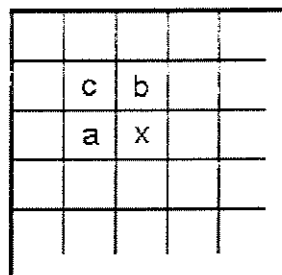


Fig. 4.6: Predição baseada nas três amostras vizinhas.

4.3 Modos de Operação dos Processos de Codificação

Existem quatro modos diferentes de operação, segundo os quais definem-se os processos distintos de codificação: a) sequencial baseado na DCT; b) progressivo baseado na DCT; c) sem perdas e; d) hierárquico.

No modo sequencial baseado na DCT, os blocos de 8x8 amostras são tomados da esquerda para a direita, e de cima para baixo. Depois de um bloco ter sido transformado pela FDCT, quantizado e preparado para a codificação entrópica, seus 64 coeficientes DCT quantizados podem ser imediatamente codificados entropicamente, e apresentados como resultado parcial dos dados comprimidos da imagem.

No modo progressivo baseado na DCT, os blocos de 8x8 amostras são codificados normalmente na mesma ordem, mas em várias varreduras da imagem. Isto é

realizado adicionando um “buffer” de memória de tamanho adequado entre o quantizador e o codificador entrópico. À medida que cada bloco é transformado pela FDCT e quantizado, os coeficientes vão sendo armazenados no “buffer”, os quais são codificados parcialmente em cada uma das varreduras múltiplas.

Na fig. 4.7 mostra-se a seqüência típica de representação das imagens na saída do decodificador no modo de operação seqüencial e progressivo.

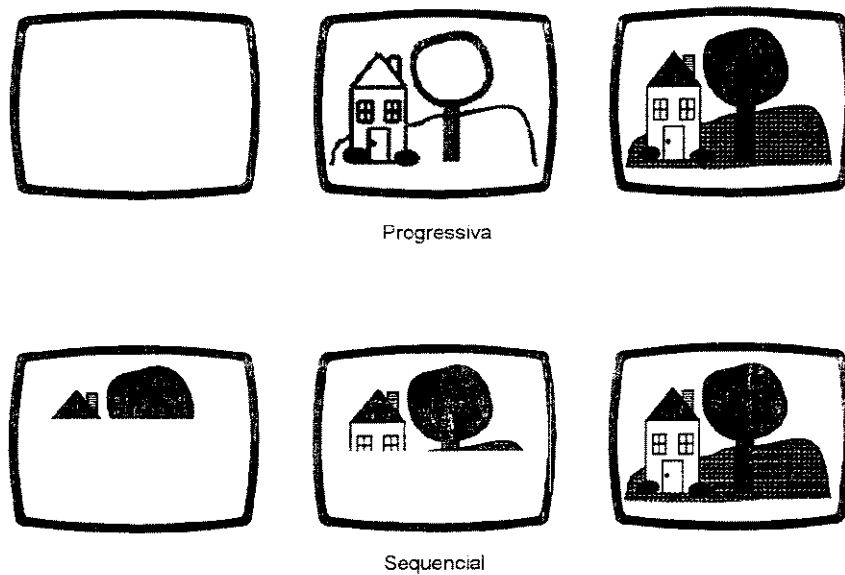


Fig. 4.7: Apresentações progressiva e seqüencial.

Existem dois procedimentos pelos quais os coeficientes quantizados armazenados no “buffer” poderiam ser parcialmente codificados dentro de uma varredura (ver fig 4.8). Primeiro, somente se necessita codificar uma banda específica de coeficientes da seqüência. Esse procedimento é chamado Seleção Espectral, devido a que cada banda contém normalmente coeficientes que ocupam a parte inferior ou a parte superior do espectro de frequência para esse bloco 8x8. Segundo, os coeficientes da banda atual não necessitam ser codificados com sua precisão máxima (de quantização) em cada varredura. Com a primeira codificação de um coeficiente, usa-se um número especificado de bits mais significativos. Nas varreduras seguintes, são usados os bits menos significativos. Este procedimento é chamado de Aproximação Sucessiva.

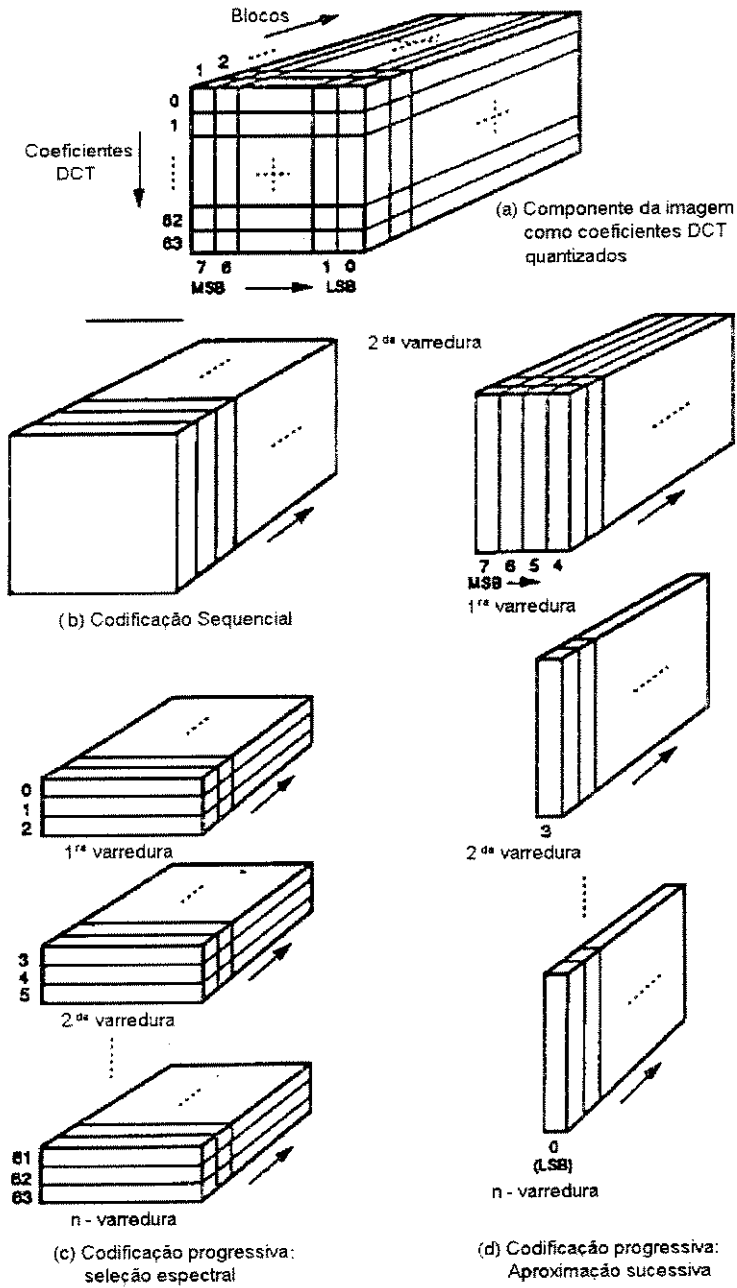


Fig. 4.8: Modos de operação dos processos de codificação baseados na DCT.

No modo hierárquico codifica-se uma imagem como uma seqüência de quadros. Estes quadros proporcionam componentes reconstruídas de referência que normalmente são necessárias para predizer os quadros seguintes. Exceto para o primeiro quadro de uma determinada componente, os quadros diferenciais codificam a diferença entre as componentes fonte e as componentes reconstruídas de referência. A codificação das diferenças pode realizar somente processos baseados na DCT, processos sem perda ou processos baseados na DCT com um processo final sem perda para cada componente. Pode-se usar os filtros de subamostragem e superamostragem para proporcionar uma pirâmide de resoluções espaciais.

A codificação sem perdas foi descrita no item 4.2.2. O grau de compressão que proporciona qualquer dos processos depende das características da própria imagem a ser comprimida, da qualidade de reconstrução desejada na aplicação e da rapidez de compressão e descompressão desejadas.

4.4 Componentes Múltiplas da Imagem

Nas seções anteriores tratou-se por simplicidade o caso de uma única componente de imagem fonte, mas a meta do proposto JPEG é lidar e controlar as imagens coloridas com componentes múltiplas.

Para um padrão de compressão genérico, o proposto deve acomodar uma ampla variedade de formatos de imagens.

4.4.1 Formato da Imagem Fonte

O modelo da imagem fonte usado pelo proposto JPEG é uma abstração de uma variedade de tipos de imagens e aplicações, e consiste somente do que é necessário para comprimir e reconstruir a informação de imagem digital. O formato do dado comprimido JPEG não especifica informação suficiente que sirva como uma representação completa

da imagem. Por exemplo, não especifica ou codifica qualquer informação da taxa de pixel, espaço de cor ou característica da aquisição da imagem.

A fig. 4.9 ilustra o modelo de imagem fonte JPEG. Uma imagem fonte contém de 1 a 255 componentes, cada componente consiste de um arranjo retangular de amostras. Uma amostra está definida como um valor inteiro positivo com a precisão de P bits e com qualquer valor na faixa $[0, 2^{P-1}]$. Todas as amostras de todas as componentes dentro da mesma imagem fonte devem ter a mesma precisão P . Para os codecs baseados na DCT, P pode ser 8 ou 12 e para os codecs preditivos P varia de 2 a 16.

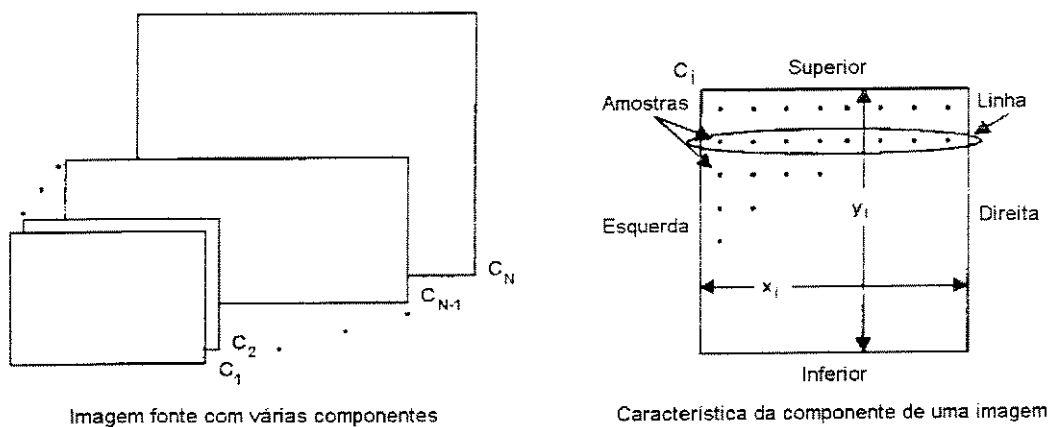


Fig 4.9: Imagem Fonte.

A i -ésima componente tem dimensões x_i e y_i . Para formatos onde algumas componentes da imagem são amostradas a taxas diferentes, as componentes podem ter dimensões diferentes. As dimensões devem ter uma relação mútua definida por H_i e V_i , os fatores de amostragem respectivos na horizontal e vertical, os quais devem ser especificados para cada componente. As dimensões totais da imagem X e Y são definidas como as máximas dimensões de x_i e y_i de todas as componentes da imagem, e podem ser qualquer valor até 2^{16} . H e V só podem ter valores inteiros de 1 a 4.

Os parâmetros de codificação para cada componente são X , Y , H_i e V_i . O decodificador reconstrói as dimensões x_i e y_i para cada componente, de acordo com as equações:

$$x_i = \left\lceil X \cdot \frac{H_i}{H_{\max}} \right\rceil \quad \text{e} \quad y_i = \left\lceil Y \cdot \frac{V_i}{V_{\max}} \right\rceil,$$

onde H_{\max} e V_{\max} são os fatores de amostragem de todas as componentes da imagem e $\lceil \cdot \rceil$ é uma função de arredondamento.

Exemplo 4.2: Considere-se uma imagem de 512 x 512 pixels definida por três componentes (C_1 , C_2 e C_3), amostradas segundo fatores dados por:

$$\begin{array}{lll} C_0: & H_0 = 4 & V_0 = 1 \\ C_1: & H_1 = 2 & V_1 = 2 \\ C_2: & H_2 = 1 & V_2 = 1 \end{array}$$

- Das equações anteriores, vemos que os tamanhos das respectivas componentes são:

$$\begin{array}{lll} C_0: & X_0 = 512 & V_0 = 256 \\ C_1: & X_1 = 256 & V_1 = 512 \\ C_2: & X_2 = 128 & V_2 = 256 \end{array}$$

4.4.2 Ordem de Codificação

Um padrão de compressão prático deve indicar como os sistemas necessitarão lidar com os dados durante o processo de descompressão. Existem aplicações em que se necessita realizar o processo de visualização ou impressão da imagem com componentes múltiplas em paralelo com o processo de descompressão. Para muitos sistemas, isto somente é viável se as componentes da imagem são interpoladas conjuntamente dentro da cadeia de dados comprimidos. Assim, o proposto JPEG define dois tipos de processamentos: interpolado e não interpolado.

Denomina-se processamento não interpolado, quando o codificador comprime todas as unidades de dados¹ de uma componente antes de começar com a componente seguinte.

No processamento interpolado, o codificador comprime as unidades de dados de cada componente alternadamente, até comprimir todas as unidades de todas as componentes. Na fig. 4.10 mostra-se um exemplo de uma imagem com três componentes de dimensões idênticas.

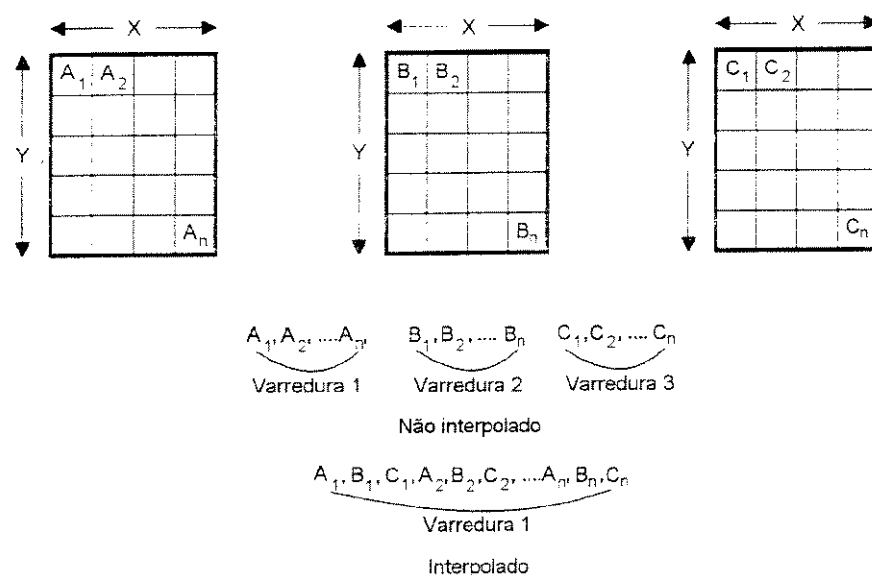


Fig. 4.10: Ordem dos dados para codificação: sem e com interpolamento.

Estes tipos de processamentos também são capazes de serem aplicados nos casos de componentes de dimensões diferentes. Nesse caso o ordenamento das unidades de informação dependerão dos fatores de amostragem no caso interpolado.

¹ Uma unidade de dados é uma amostra nos processos sem perda, e um bloco de 8x 8 amostras nos processos baseados na DCT.

Exemplo 4.3: Considere-se o exemplo anterior no qual as três componentes de imagem (C_0 , C_1 e C_2) são amostradas segundo fatores de amostragem diferentes (ver fig. 4.11).

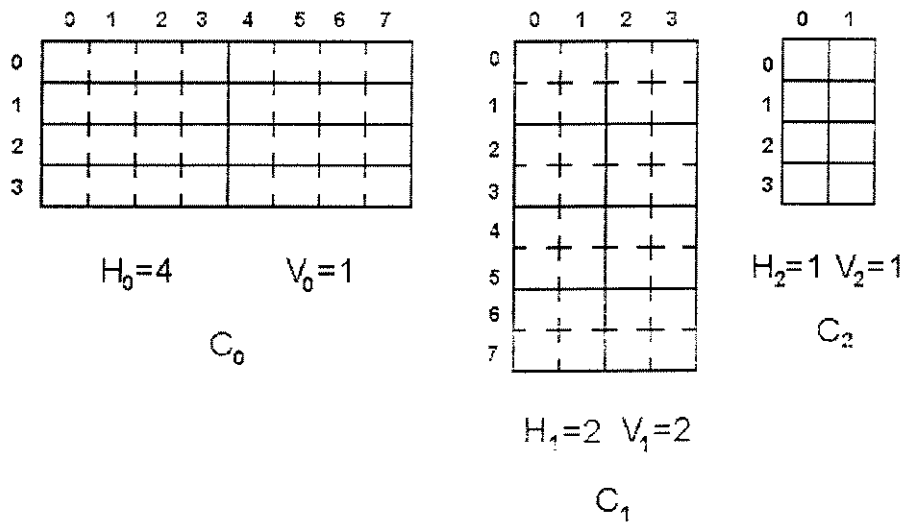


Fig. 4.11: Ordem da codificação interpolada das componentes com dimensões diferentes.

- O ordenamento das unidades de dados em forma interpolada será:

$$MCU_1 = C_0[0,0], C_0[0,1], C_0[0,2], C_0[0,3], C_1[0,0], C_1[0,1], C_1[1,0], C_1[1,1] \text{ e } C_2[0,0]$$

$$MCU_2 = C_0[0,4], C_0[0,5], C_0[0,6], C_0[0,7], C_1[0,2], C_1[0,3], C_1[1,2], C_1[1,3] \text{ e } C_2[0,1]$$

Sendo que MCU (Minimum Coding Unit) é a mínima unidade a ser codificada.

A sequência de dados que entram no codificador será dada por :

$$MCU_1, MCU_2, \dots$$

Observa-se que geralmente as unidades de dados são ordenadas da esquerda para a direita, e de cima para baixo.

4.5 Tabelas Múltiplas

Adicionalmente ao controle do processamento interpolado ou não, os codecs JPEG devem controlar a aplicação da tabela apropriada de dados às componentes. A mesma tabela de quantização e a mesma tabela de codificação entrópica devem ser usadas para codificar todas as amostras dentro de uma componente.

Os decodificadores JPEG podem armazenar até quatro tabelas diferentes de quantização e de codificação entrópica. O decodificador do modelo básico JPEG é uma exceção, somente pode armazenar até duas tabelas de codificação entrópica.

O controle é necessário para se comutar entre as diferentes tabelas durante a descompressão de uma varredura contendo componentes múltiplas, e aplicar a tabela apropriada à componente. A fig. 4.12 ilustra um exemplo de controle de comutação para as tabelas e que deve ser usado conjuntamente com as componentes.

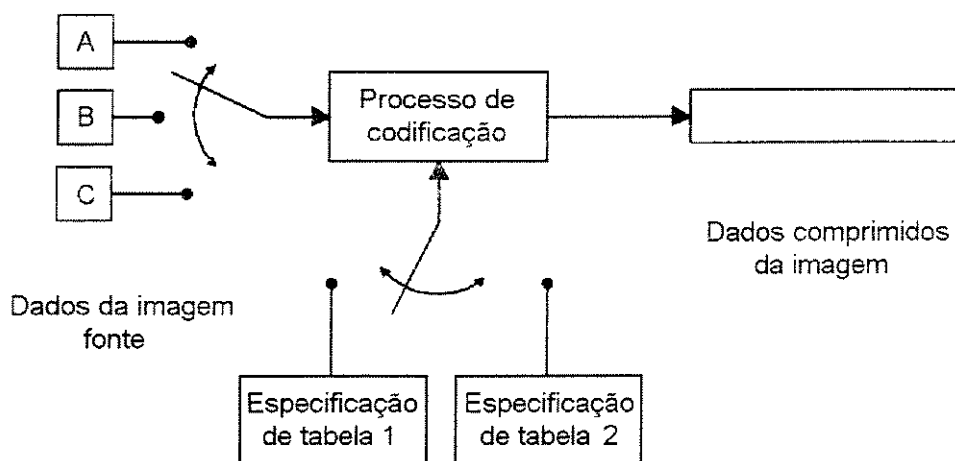


Fig. 4.12: Controle de comutação para as tabelas e as componentes.

4.6 Resumo das Características Essenciais dos Processos de Codificação do Padrão JPEG

- Processo Básico
 - Processo baseado na DCT.
 - Fonte de imagem: amostras de 8 bits em cada componente.
 - Seqüencial.
 - Codificação de Huffman.
 - Os codificadores deverão processar as varreduras com 1, 2, 3 e 4 componentes.
 - Varreduras com e sem interpolação.
- Processo Estendido
 - Processo baseado na DCT.
 - Fonte de imagem: amostras de 8 ou 12 bits.
 - Seqüencial ou progressiva.
 - Codificação de Huffman ou aritmética.
 - Os codificadores deverão processar as varreduras com 1, 2, 3 e 4 componentes.
 - Varreduras com e sem interpolação.
- Processo sem Perda
 - Processo preditivo.
 - Fonte de imagem: amostras de P bits ($2 \leq P \leq 16$).
 - Seqüencial.
 - Codificação de Huffman.
 - Os codificadores deverão processar as varreduras com 1, 2, 3 e 4 componentes.
 - Varreduras com e sem interpolação.
- Processo Hierárquico
 - Vários quadros (não diferenciais e diferenciais).
 - Utilização dos processos baseados na DCT estendida ou sem perda.
 - Os codificadores deverão processar as varreduras com 1, 2, 3 e 4 componentes.
 - Varreduras com e sem interpolação.

Capítulo 5

Simulação e Resultados

5.1 Introdução

Neste capítulo, são explicados com mais detalhes os diversos processos do sistema JPEG básico, cujo comportamento é estudado através de simulações em computador com programas desenvolvidos em linguagem C.

A simulação foi implementada em software dada a sua importância de servir como base para futuras adaptações e otimizações na codificação de imagens paradas. Além disso, o modelo básico é a base para a codificação de imagens em movimento segundo o padrão MPEG.

Também são apresentadas as tabelas de dados pré-estabelecidas utilizados nas diversas etapas, os sinais de testes usados como imagens fonte, as ferramentas empregadas que facilitaram o tratamento das imagens e a avaliação do sistema, e finalmente os resultados dos testes objetivos e subjetivos do sistema implementado.

5.2 Espaço de Cor

Geralmente a fonte de imagens encontra-se na forma de componentes R (red), G (green) e B (blue). Sabe-se que estas são altamente correlacionadas uma com a outra. Além disso, o sistema visual humano responde diferentemente às componentes de luminância e cromaticidade. Assim, para reduzir a correlação e explorar essa diferença no sistema visual humano, as componentes RGB são convertidas para o espaço de cor YUV (a componente Y corresponde à luminância, enquanto que, U e V são os sinais diferença

de cor à cromaticidade), através de uma transformação linear. A conversão correspondente escrita em forma de matriz é dada por :

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.5870 & 0.114 \\ -0.1687 & -0.3313 & 0.5 \\ 0.5 & -0.4187 & -0.0813 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

A conversão inversa é dada por :

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.000 & 0 & 1.402 \\ 1.000 & -0.34414 & -0.714141 \\ 1.000 & 1.772 & 0 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix}$$

No espaço de cor YUV, a maioria das componentes de alta frequência são concentradas na componente Y. Além disso, o sistema visual é menos sensível às componentes de alta frequência da cromaticidade do que às componentes de alta frequência da luminância. Para explorar essas características, num sistema de compressão de vídeo as componentes de cromaticidade são filtradas por um filtro passa-baixas e subamostradas por um fator de dois ao longo das dimensões horizontais e verticais, produzindo componentes de cromaticidade que correspondem a um quarto da resolução espacial da componente de luminância.

Um esquema da utilização dos sinais teste da imagem em componentes que é empregado para a simulação do sistema básico JPEG é mostrado na fig. 5.1.

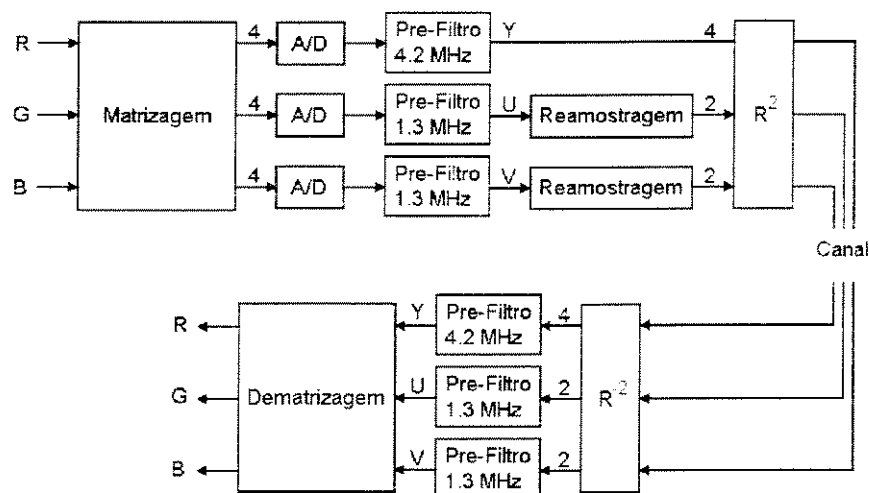


Fig. 5.1: Esquema de conversão do espaço RGB 4:4:4 ao espaço YUV 4:2:2.

5.3 Modelo JPEG Básico

Para a implementação das diferentes etapas do codec JPEG básico utilizaram-se as Recomendações das normas do ISO/IEC 10918 - 1 [25], especificamente os anexos A, B, C, E, F e K:

- A (Definições matemáticas).
- B (Formato dos dados comprimidos).
- C (Especificação das tabelas de Huffman).
- E (Procedimentos de controle do codificador e decodificador).
- F (Modo de operação seqüencial baseado na DCT).
- K (Exemplos e diretrizes).

Para facilitar o entendimento, as tabelas dos anexos utilizadas na implementação são reproduzidas neste capítulo. Cabe ressaltar que para a implementação do software simulador tentou-se seguir o mais fielmente possível os procedimentos da forma especificada pelos gráficos de fluxo dos anexos mencionados.

Uma orientação mais clara das diversas etapas envolvidas é conseguida mostrando-se na fig. 5.2 o diagrama simplificado do codec (codificador e decodificador) JPEG básico novamente, e analisando-se então essas etapas.

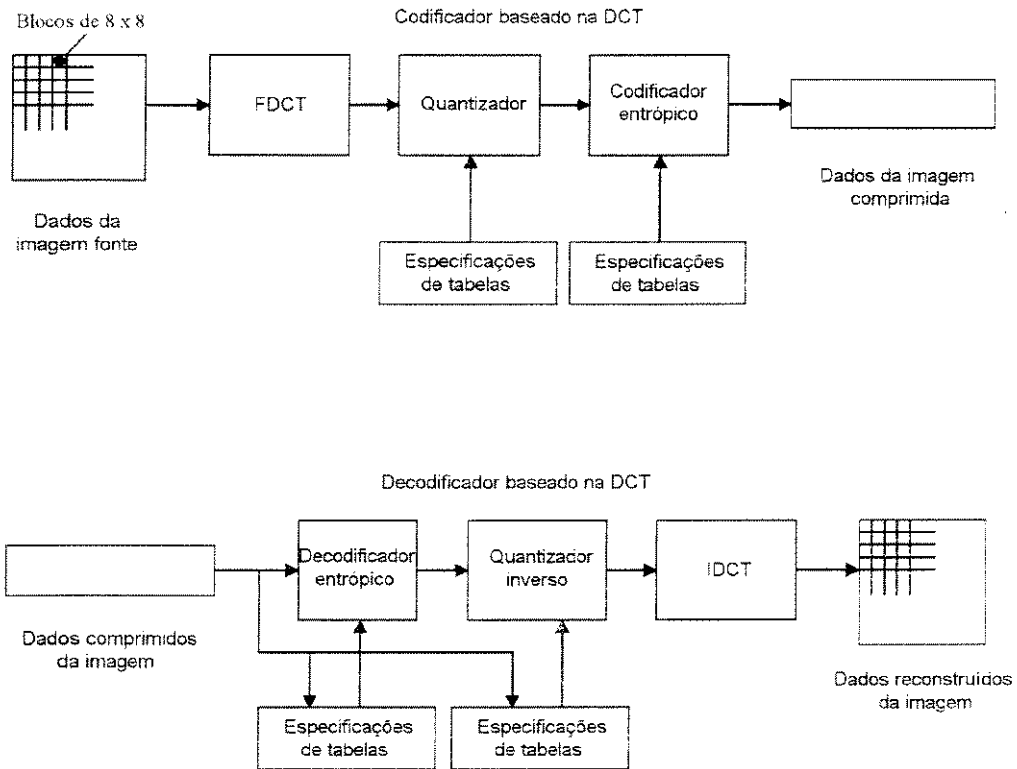


Fig. 5.2: Diagrama simplificado do codec JPEG básico.

5.3.1 Transformadas : FDCT e IDCT

Para simplificar a complexidade da transformada, as amostras da imagem geradas no intervalo $[0 a 2^P-1]$ devem ser previamente deslocadas para uma representação com sinal no intervalo $[-2^{P-1}, 2^{P-1}-1]$, subtraindo-se 2^{P-1} dos valores das amostras, onde P é a precisão da amostra. Depois do processo da decodificação, um deslocamento inverso restaura as amostras para a representação sem sinal adicionando-se 2^{P-1} , ajustando dessa forma novamente os valores na faixa de $[0 a 2^{P-1}]$.

As amostras previamente deslocadas são agrupadas em blocos de 8x8 amostras. Cada bloco transforma-se por meio da FCDT num conjunto de 64 valores denominados coeficientes DCT. A fig. 5.3 apresenta uma componente de imagem que foi dividida em blocos de 8x8 para os cálculos da FDCT, bem como a orientação das amostras dentro de um bloco.

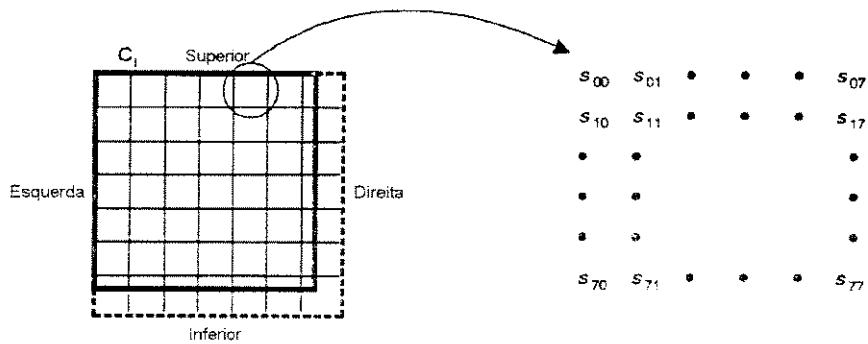


Fig. 5.3: Partição e orientação dos blocos de 8x8 amostras.

A definição funcional teórica da FDCT (Forward Discrete Cosine Transform) e da IDCT (Inverse Discrete Cosine Transform) bidimensionais 8x8 pode ser escrita em função dos pixel s_{yx} e dos coeficientes transformados no domínio da frequência S_{vu} .

FDCT:

$$S_{vu} = \frac{1}{4} C_u C_v \sum_{x=0}^7 \sum_{y=0}^7 s_{yx} \cos \left[(2x + 1) \frac{u\pi}{16} \right] \cos \left[(2y + 1) \frac{v\pi}{16} \right] \quad (5.3)$$

IDCT:

$$s_{yx} = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 C_u C_v S_{vu} \cos \left[(2x + 1) \frac{u\pi}{16} \right] \cos \left[(2y + 1) \frac{v\pi}{16} \right] \quad (5.4)$$

onde, C_u e C_v igual a $1/\sqrt{2}$ para $u,v=0$ e 1 nos demais casos.

Dos 64 coeficientes DCT refere-se a um deles como o coeficiente DC (o de mais baixa frequência, ou seja $u=v=0$) e aos outros 63 como os coeficientes AC (u ou $v \neq 0$).

5.3.2 Quantização

Para favorecer a compressão, cada um dos 64 coeficientes DCT são representados com uma precisão não muito grande, mas suficiente para obter uma imagem de qualidade desejável.

Assim, o método base JPEG utiliza uma quantização uniforme do tipo “midstep” dada por uma tabela de 64 valores, a qual deve ser especificada segundo a aplicação.

A quantização é definida como:

$$Sq_{vu} = \text{inteiro mais próximo} \left[\frac{S_{vu}}{Q_{vu}} \right], \quad (5.5)$$

onde Sq_{vu} é o coeficiente quantizado da DCT, normalizado de acordo com o tamanho do passo do quantizador.

No decodificador suprime-se essa normalização por meio da quantização inversa dada por:

$$R_{vu} = Sq_{vu} \cdot Q_{vu}, \quad (5.6)$$

onde R_{vu} é o valor reconstruído de S_{vu} .

As tabelas 5.1 e 5.2 são dois exemplos de tabelas de quantização de luminância e crominância.

A fig. 5.4 ilustra a relação entre as amostras, os coeficientes da DCT e os coeficientes quantizados.

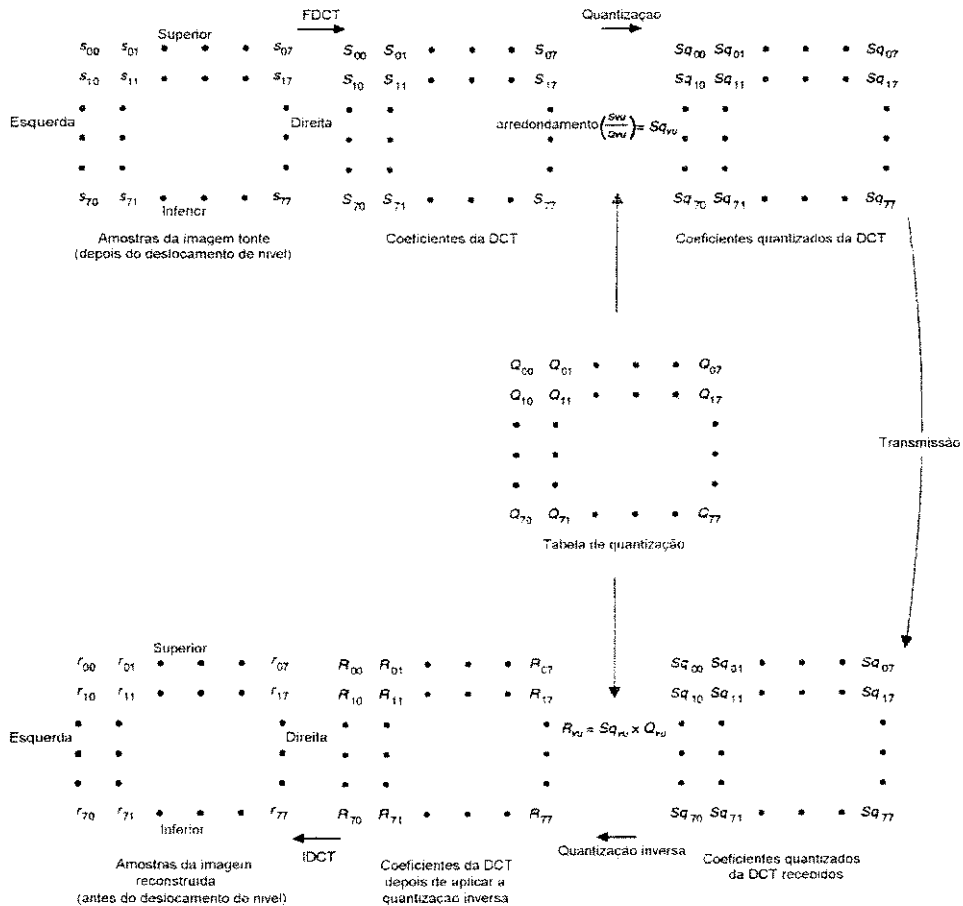


Fig. 5.4: Relação entre os blocos de amostras 8x8 e os coeficientes da DCT.

5.3.3 Modelo de Codificação

Depois da quantização, e como preparação para a codificação entrópica, o coeficiente DC quantizado Sq_{00} é tratado separado dos 63 coeficientes AC. O valor que será codificado é a diferença (DIFF) entre o coeficiente DC quantizado do bloco atual (DC_i , designado também como o coeficiente Sq_{00} do i -ésimo bloco) e o do bloco precedente da mesma componente (DC_{i-1}):

$$\text{DIFF}_i = \text{DC}_i - \text{DC}_{i-1}.$$

Isso é feito porque o coeficiente DC representa o valor médio das 64 amostras, e também se pressupõe que exista usualmente uma forte correlação entre os coeficientes DC de blocos adjacentes.

Por outro lado, os coeficientes AC quantizados são ordenados dentro de uma seqüência unidimensional “zigzag” (ver fig. 5.5), para favorecer a colocação dos valores dos coeficientes mais significativos (de mais baixa freqüência) primeiro, seguidos pelos menos significativos (de mais alta freqüência). Isso favorece a formação de filas ou correntes de zeros. Por isso, uma codificação vantajosa em termos de compressão pode ser obtida usando-se a técnica de Run-Length que codifica o número de zeros consecutivos da corrente antes do próximo coeficiente diferente de zero. Além disso, quando os últimos coeficientes transformados são todos de valores nulos, tais coeficientes são codificados com o código EOB (fim do bloco).

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 0 | 1 | 5 | 6 | 14 | 15 | 27 | 28 |
| 2 | 4 | 7 | 13 | 16 | 26 | 29 | 42 |
| 3 | 8 | 12 | 17 | 25 | 30 | 41 | 43 |
| 9 | 11 | 18 | 24 | 31 | 40 | 44 | 53 |
| 10 | 19 | 23 | 32 | 39 | 45 | 52 | 54 |
| 20 | 22 | 33 | 38 | 46 | 51 | 55 | 60 |
| 21 | 34 | 37 | 47 | 50 | 56 | 59 | 61 |
| 35 | 36 | 48 | 49 | 57 | 58 | 62 | 63 |

Fig. 5.5: Seqüência zigzag dos coeficientes quantizados da DCT.

5.3.4 Codificação Entrópica

Ambos os códigos de blocos dos modelos DPCM e Run-Length são codificados entropicamente (Huffman ou Aritmética).

O método JPEG base usa o codificador de Huffman, o qual para comprimir símbolos de informação, cria códigos curtos para símbolos que ocorrem frequentemente e códigos compridos para símbolos que ocorrem ocasionalmente.

Para o codificador de Huffman requer-se que sejam especificados, tanto no codificador como no decodificador, um ou mais conjuntos de tabelas de códigos de Huffman. As tabelas de Huffman podem ser pré-definidas e usadas como “default”, ou computadorizadas especificamente para uma dada imagem num passo inicial prévio à compressão.

Estrutura dos Códigos DC

Cada uma das magnitudes das diferenças (DIFF) está relacionada a uma das doze categorias denominadas SSSS (conforme tabela 5.3), as quais correspondem um código de Huffman (ver anexo A). Para identificar exatamente a diferença produzida nessa categoria adiciona-se um campo de bits representado pelos bits mais significativos (b.m.s.) do valor de DIFF (o número de bits extra é dado por SSSS). Isto é:

$$\text{DIFF} \rightarrow \text{SSSS} \rightarrow [\text{COD.HUFF.} + (\text{b.m.s.}) \text{DIFF}]$$

Tab. 5.3: Categorias das magnitudes da diferença.

| SSSS | Valores de DIFF |
|------|---------------------------|
| 0 | 0 |
| 1 | -1,1 |
| 2 | -3,-2,2,3 |
| 3 | -7...-4,4...7 |
| 4 | -15...-8,8...15 |
| 5 | -31...-16,16...31 |
| 6 | -63...32,32...63 |
| 7 | -127...-64,64...127 |
| 8 | -255...-128,128...255 |
| 9 | -511...-256,256...511 |
| 10 | -1023...-512,512...1023 |
| 11 | -2047...-1024,1024...2047 |

Estrutura dos Códigos AC

As magnitudes dos coeficientes AC diferentes de zero estão relacionadas a uma das dez categorias denominadas SSSS (conforme tabela 5.4). Para cada corrente de zeros, e para cada categoria lhes corresponde um código (ver anexo A). Adiciona-se um campo de bits ao código de Huffman, representado pelos bits mais significativos (b.m.s.) do coeficiente e pela amplitude exata do próximo coeficiente. Isto é:

AC → SSSS → (CORRENTE DE ZEROS, SSSS) → [COD.HUFF. + (b.m.s.) AC].

Tab. 5.4: Categorias relacionadas aos coeficientes AC

| SSSS | Coeficientes AC |
|------|-------------------------|
| 1 | -1,1 |
| 2 | -3,-2,2,3 |
| 3 | -7...-4,4...7 |
| 4 | -15...-8,8...15 |
| 5 | -31...-16,16...31 |
| 6 | -63...32,32...63 |
| 7 | -127...-64,64...127 |
| 8 | -255...-128,128...255 |
| 9 | -511...-256,256...511 |
| 10 | -1023...-512,512...1023 |

5.4 Simulação e Avaliação do Sistema JPEG Básico.

Em geral, a avaliação do desempenho de um sistema ou equipamento de comunicações é realizada através do uso de aparelhos de testes que medem os parâmetros que caracterizam tal sistema ou equipamento. Naturalmente, no caso de desenvolvimentos de projetos, tais medidas objetivas são imprescindíveis, porém é também possível obter-se uma avaliação do desempenho através de simulações do sistema.

Geralmente as simulações são realizadas com ajuda de um computador que permite testar muitos parâmetros com diferentes condições aplicadas ao equipamento físico real, a fim de se prever o comportamento do mesmo nessas situações. Assim, a vantagem das simulações em relação à implementação física real é evidente quando é grande o número de opções que se têm para se construir o equipamento.

A confiabilidade dos resultados obtidos através de simulações é limitada pela fidelidade do modelo usado para representar o equipamento real. Comumente uma vez que é impossível modelar exatamente a realidade, as simulações têm em geral um caráter de apoio ao projeto, sendo que o desempenho de fato deve ser obtido no equipamento fisicamente implementado.

5.4.1 Plataforma de Trabalho Implementada

A plataforma de trabalho implementada para a simulação do codec JPEG básico transforma as fontes de imagens nas componentes Y, U e V numa imagem comprimida JPEG e inverte a operação (ver fig. 5.6).

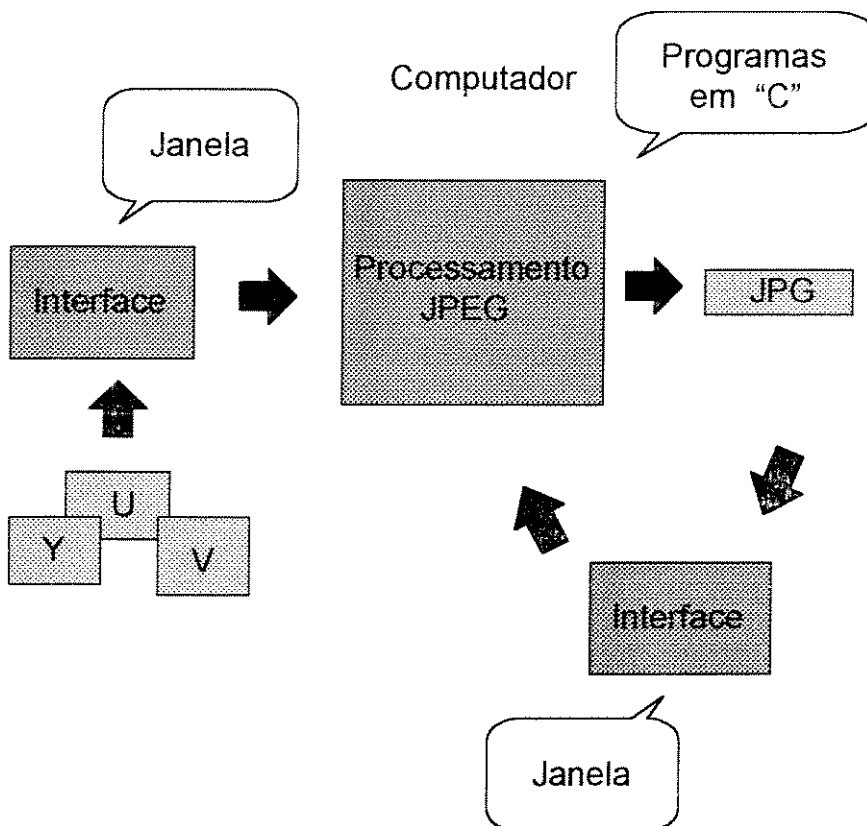


Fig. 5.6: Sequência de processamento da plataforma implementada.

Essa plataforma está conformada por duas janelas: uma para codificação e outra para a decodificação, permitindo uma manipulação fácil pelo usuário, pois ele só necessita preencher os requisitos solicitados nas janelas, tais como: largura e altura máxima da imagem, nome das componentes da imagem a ser codificada (com extensões .Y, .U e .V) com as respectivas frequências de amostragem, modo de operação, opção para se conhecer a entropia resultante ou não (ver fig. 5.7 para a codificação), bem como o nome da imagem comprimida a ser decodificada (ver fig. 5.8 para a decodificação).

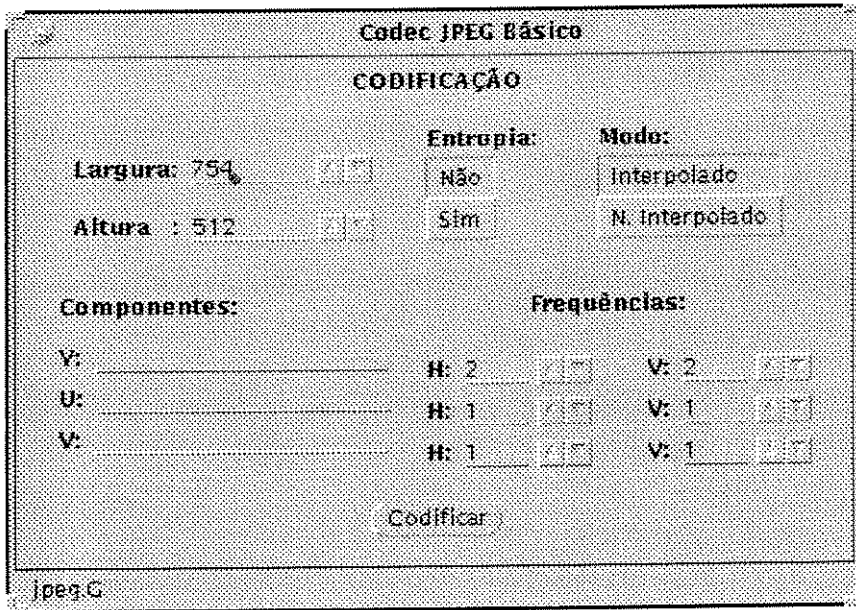


Fig. 5.7: Janela da plataforma de codificação JPEG básico.

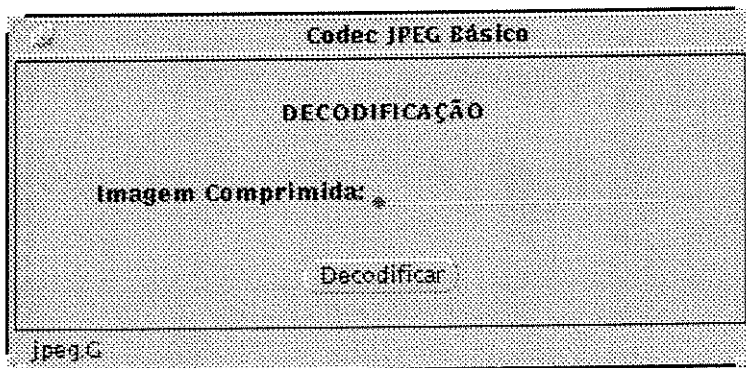


Fig. 5.8: Janela da plataforma de decodificação JPEG básico.

5.4.2 Sinais de Testes

Nessa simulação, utilizam-se dois tipos de sinais: estatísticos e determinísticos. Os sinais estatísticos usados são obtidos pela digitalização de imagens padrão da SMPTE (Society of Motion Pictures and Television Engineers). Cada imagem está conformada por três quadros de 512x512 pixels de 8 bits, a uma taxa aproximada de 10 MHz, correspondentes às componentes primárias R, G e B. Essas imagens formam um conjunto representativo das cenas comumente encontradas. Das quinze imagens padrão da SMPTE foram utilizadas as imagens que apresentam maiores variações de detalhes, de luz e ambiente, como SMPTE01 (praia), SMPTE02 (sala clara), SMPTE08 (sala escura) e SMPTE15 (cozinha). Todas estas são apresentadas na fig. 5.9.

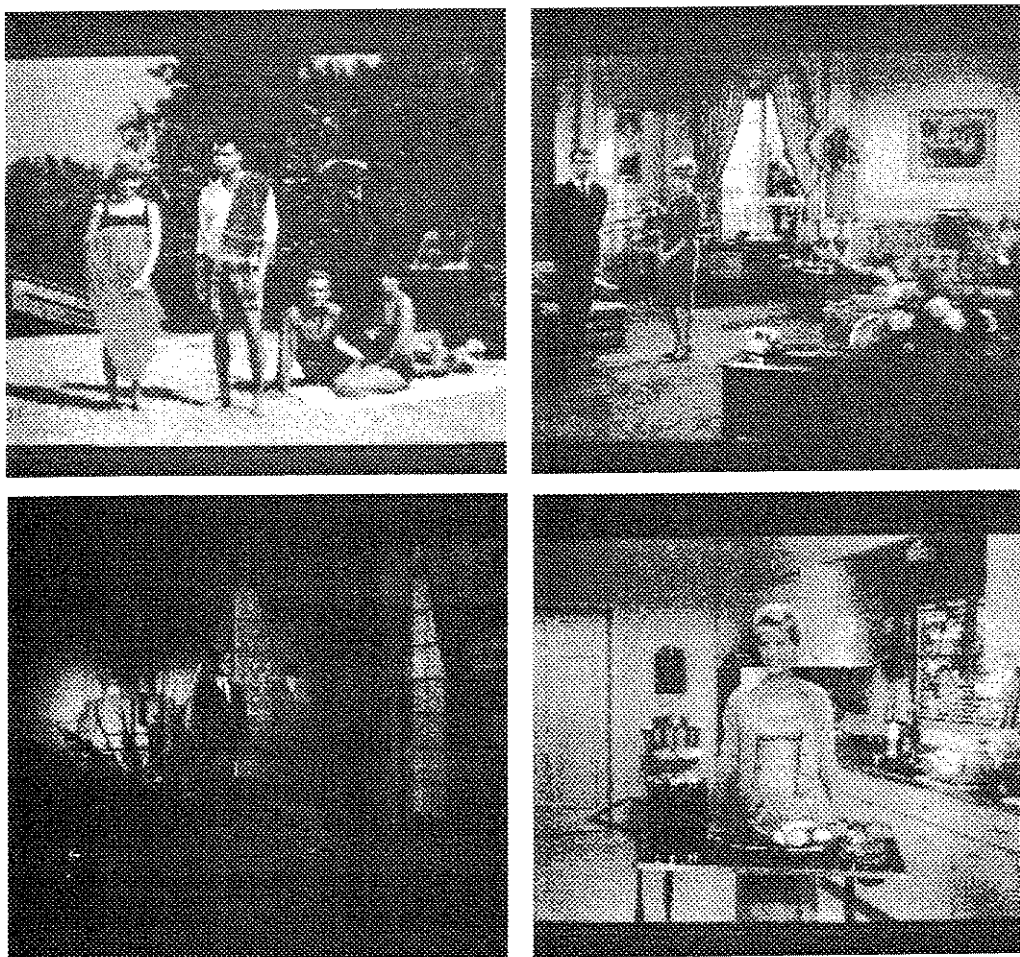


Fig.5.9: Sinais testes do tipo estatísticos: smpte01, smpte02, smpte08 e smpte15.

Os sinais determinísticos usados foram aqueles padronizados pelo CCIR do tipo gerado eletronicamente para diversos testes em sistema analógicos tal como: Barras Coloridas (branco, amarelo, magenta, ciano, verde, vermelho, azul e preto), de 755x512 pixels de 8 bits, a uma taxa de 4fsc (ver fig. 5.10), onde fsc é a frequência da subportadora de cor.

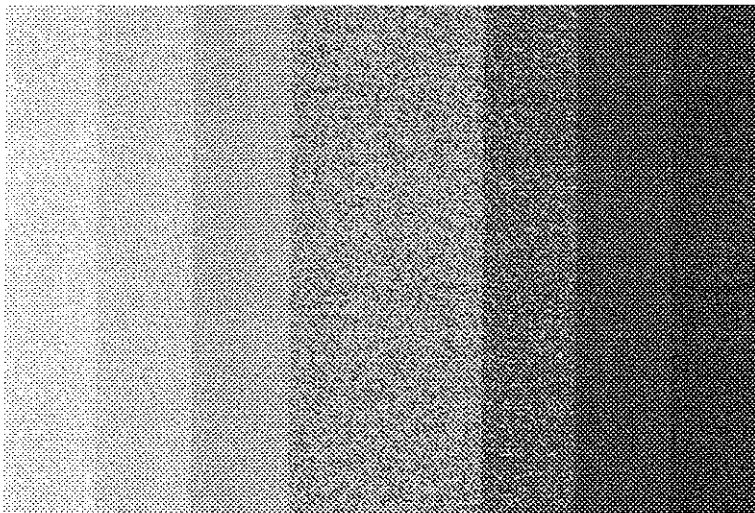


Fig.5.10: Sinal teste do tipo determinístico: barras coloridas.

As componentes da imagem original no espaço de cor RGB são previamente convertidas ao espaço de cor YUV e subamostradas na forma 4:2:2, a fim de explorar ao máximo a redução de redundância e limitações do sistema visual humano. Estas últimas são consideradas os sinais de testes que são processados pelo sistema simulado.

5.4.3 Capacidade de Compressão

A capacidade de compressão do sistema JPEG Básico simulado, em termos de memória ocupada pela imagem original e comprimida, é mostrada na tabela 5.5:

Tab. 5.5: Memória ocupada pelos sinais originais e comprimidos.

| Sinais de Teste | No espaço RGB Total (bytes) | No espaço YUV Total (bytes) | Comprimido Total (bytes) |
|-----------------|-----------------------------|-----------------------------|--------------------------|
| Smpte01 | 786432 | 393216 | 21750 |
| Smpte02 | 786432 | 393216 | 21048 |
| Smpte08 | 786432 | 393216 | 13267 |
| Smpte15 | 786432 | 393216 | 21232 |
| Barras | 1159680 | 579072 | 10263 |

A compressão alcançada é alta e a imagem recuperada mantém um bom nível de fidelidade com relação à imagem original. Nota-se que a taxa de compressão depende muito da quantidade de detalhes da imagem. Assim a imagem determinística apresenta maior facilidade para reduzir as redundâncias, devido à escassez de detalhes e a que esse sinal não apresenta variações na vertical.

5.4.4 Avaliação Objetiva

Essa avaliação é importante para a análise do desempenho do sistema, e é feita através da medida de certos parâmetros relevantes (entropia, erro quadrático médio, relação sinal/ruído, etc) que traduzem em resultados numéricos, e portanto quantitativos, o comportamento do sistema durante sua operação.

A tabela 5.6 mostra um resumo dos resultados da avaliação objetiva do sistema com o erro quadrático médio e a relação sinal/ruído.

Tab. 5.6: MSE e SNR das componentes da imagem.

| Sinais de Teste | Y | | U | | V | |
|-----------------------|-----|-------------|-----|-------------|-----|-------------|
| | MSE | SNR (dB) | MSE | SNR (dB) | MSE | SNR (dB) |
| Smpte01 | 32 | 33.08 | 7 | 39.68 | 8 | 39.10 |
| Smpte02 | 35 | 32.69 | 6 | 40.35 | 8 | 39.10 |
| Smpte08 | 15 | 36.37 | 8 | 39.10 | 7 | 39.68 |
| Smpte15 | 43 | 31.80 | 6 | 40.35 | 12 | 39.10 |
| Barras | 8 | 39.10 | 5 | 41.14 | 5 | 41.14 |

Mostram-se no apêndice as frequências computadas de cada comprimento de palavra dos coeficientes codificados, para o cálculo da entropia das diversas componentes. Cabe ressaltar que a entropia indica o número mínimo de bit/amostra necessários para transmitir os coeficientes transformados do quadro, de modo que deve satisfazer as condições de limites de largura de faixa requerida pela frequência de amostragem, quais sejam, 34 ou 68 Mbit/s. As entropias resultantes são dadas na tabela 5.7.

Tab. 5.7: Entropia das componentes da imagem.

| Sinais de Teste | Y | | U | | V | |
|-----------------------|------------|-------|------------|-------|------------|-------|
| | H (bpp) | (Mbs) | H (bpp) | (Mbs) | H (bpp) | (Mbs) |
| Smpte01 | 0.85 | 8.5 | 0.65 | 6.5 | 0.66 | 6.6 |
| Smpte02 | 0.86 | 8.6 | 0.65 | 6.5 | 0.62 | 6.2 |
| Smpte08 | 0.84 | 8.4 | 0.55 | 5.5 | 0.43 | 4.3 |
| Smpte15 | 0.86 | 8.6 | 0.66 | 6.6 | 0.57 | 5.7 |
| Barras | 0.54 | 7.7 | 0.56 | 8.0 | 0.52 | 7.4 |

5.4.5 Avaliação Subjetiva

Esta avaliação é feita após o processamento dos sinais pelo sistema, comparando a imagem original com a recuperada, tentando-se seguir os critérios estabelecidos pelo CCIR Rec. 500-1 Vol. XI [26]. As limitações de nossos recursos não permitem a reprodução das condições ideais. As tabelas recomendadas para testes subjetivos são apresentadas nas tabelas 5.8 e 5.9.

Tab. 5.8: Escala de 5 graus para a medida subjetiva de qualidade ou de degradação.

| Escala de 5 graus | |
|-------------------|-------------------------------------|
| Qualidade | Degradação |
| 5. Excelente | 5. Imperceptível |
| 4. Boa | 4. Perceptível mas não perturbadora |
| 3. Regular | 3. Levemente perturbadora |
| 2. Ruim | 2. Perturbadora |
| 1. Péssima | 1. Muito perturbadora |

Tab. 5.9: Escala de comparação para a medida subjetiva da qualidade do sinal recuperado.

| Escala de comparação | |
|----------------------|---------------------|
| +3 | Muito melhor |
| +2 | Melhor |
| +1 | Ligeiramente melhor |
| 0 | A mesma |
| -1 | Ligeiramente pior |
| -2 | Pior |
| -3 | Muito pior |

Mostram-se nas figs. 5.10 a 5.14, as imagens de testes originais e recuperadas a fim de ter uma comparação e determinação da fidelidade de imagem alcançada.

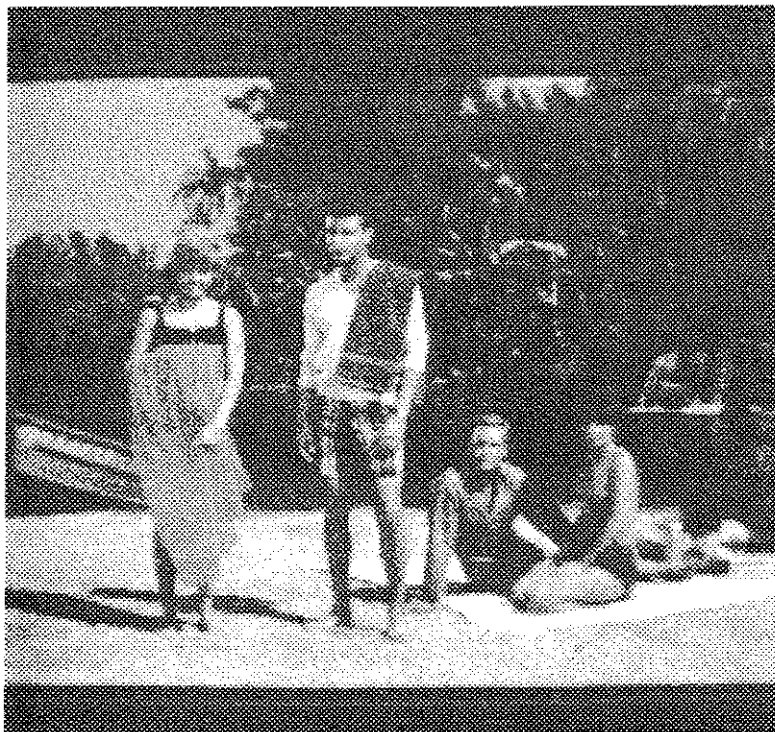


Fig.5.11: Imagem smpte01 original e recuperada .

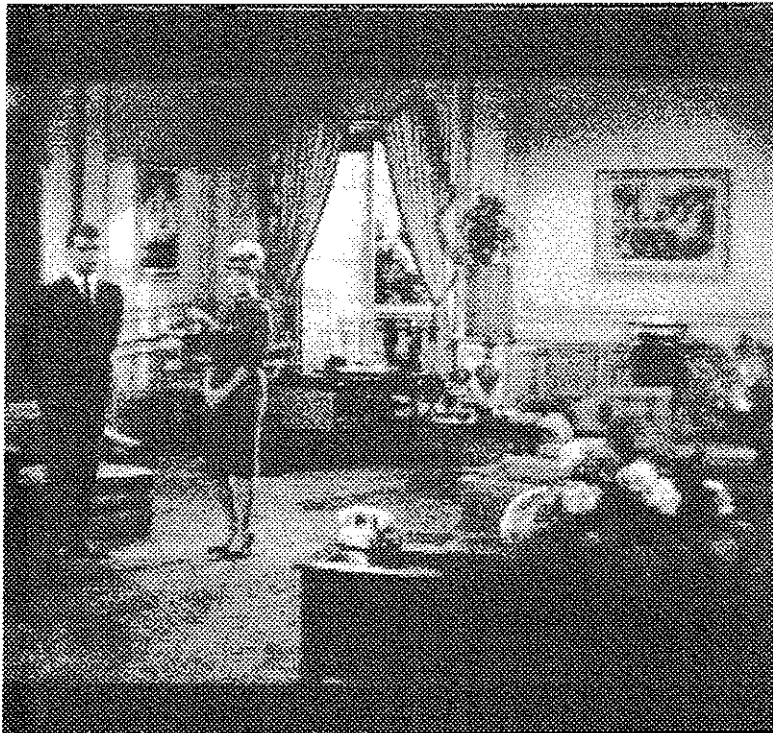
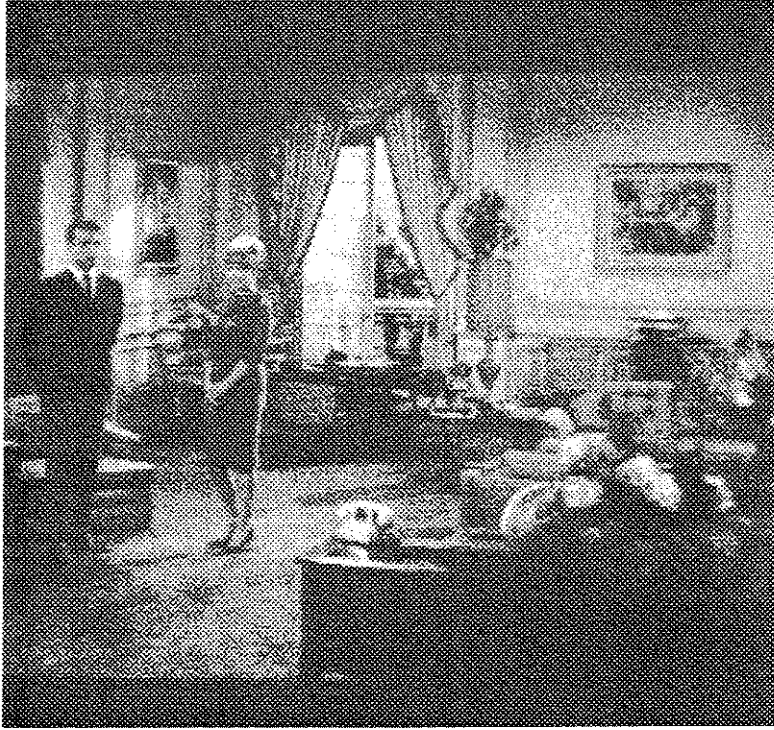


Fig.5.12: Imagem smpte02 original e recuperada.



Fig.5.13: Imagem smpte08 original e recuperada.



Fig.5.14: Imagem smpte15 original e recuperada.

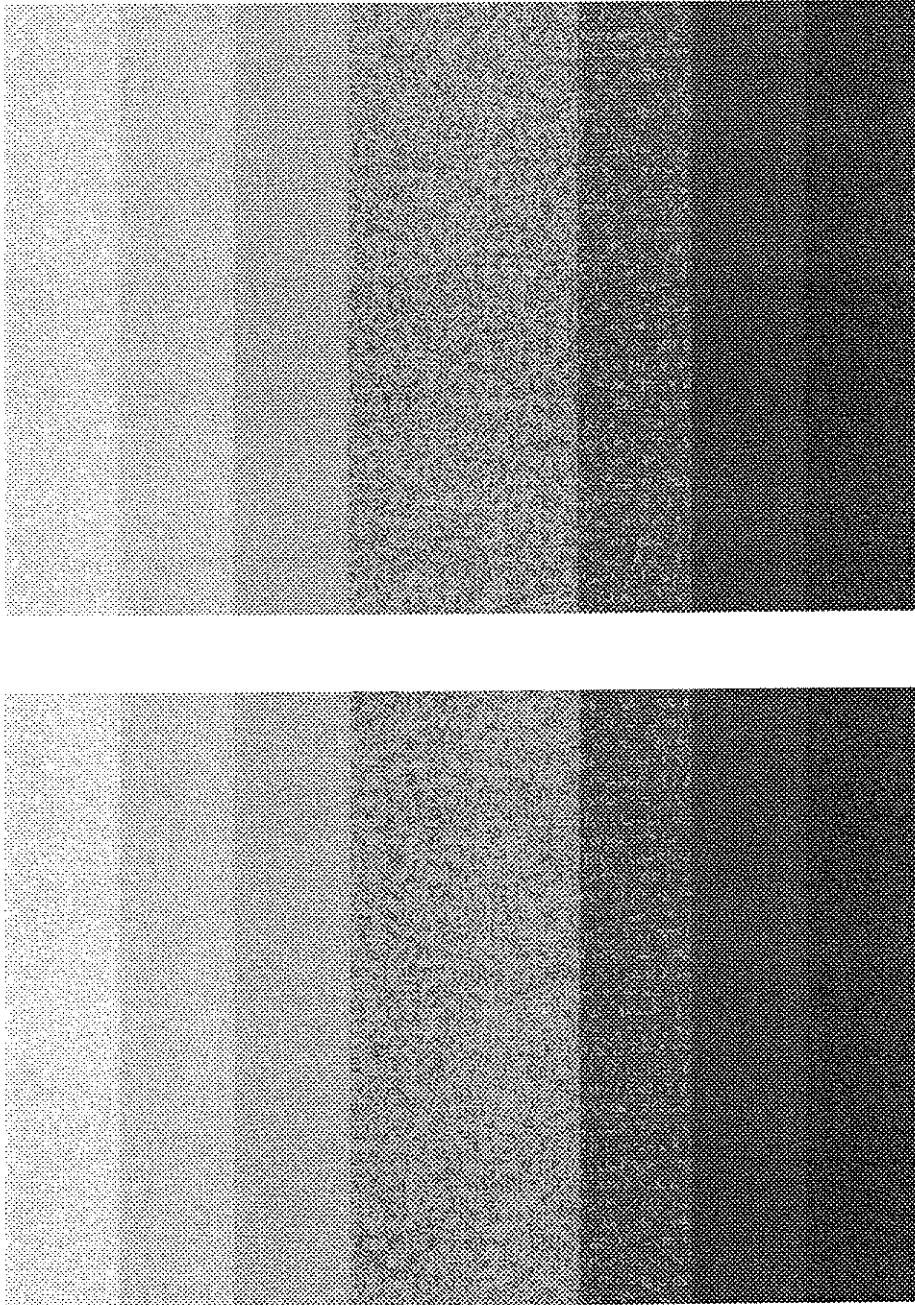


Fig.5.15: Imagem barras coloridas original e recuperada

Seis observadores (colegas de curso na FEE/UNICAMP) avaliaram a semelhança entre as imagens (original e recuperada), conforme sugerido pelo CCIR. A média do resultado da avaliação feita é apresentada na Tabela 5.10.

Tabela 5.10: Resultado da avaliação subjetiva.

| Imagem | Nota |
|---------|------|
| Barras | 5 |
| Smpte01 | 5 |
| Smpte02 | 5 |
| Smpte08 | 5 |
| Smpte15 | 5 |

A seguir mostram-se os histogramas correspondentes às componentes de imagem (originais e recuperadas), que foram obtidos através do “workspace” construído no KHOROS (ver fig. 5.16), com o objetivo de avaliar as mudanças de intensidades entre elas.

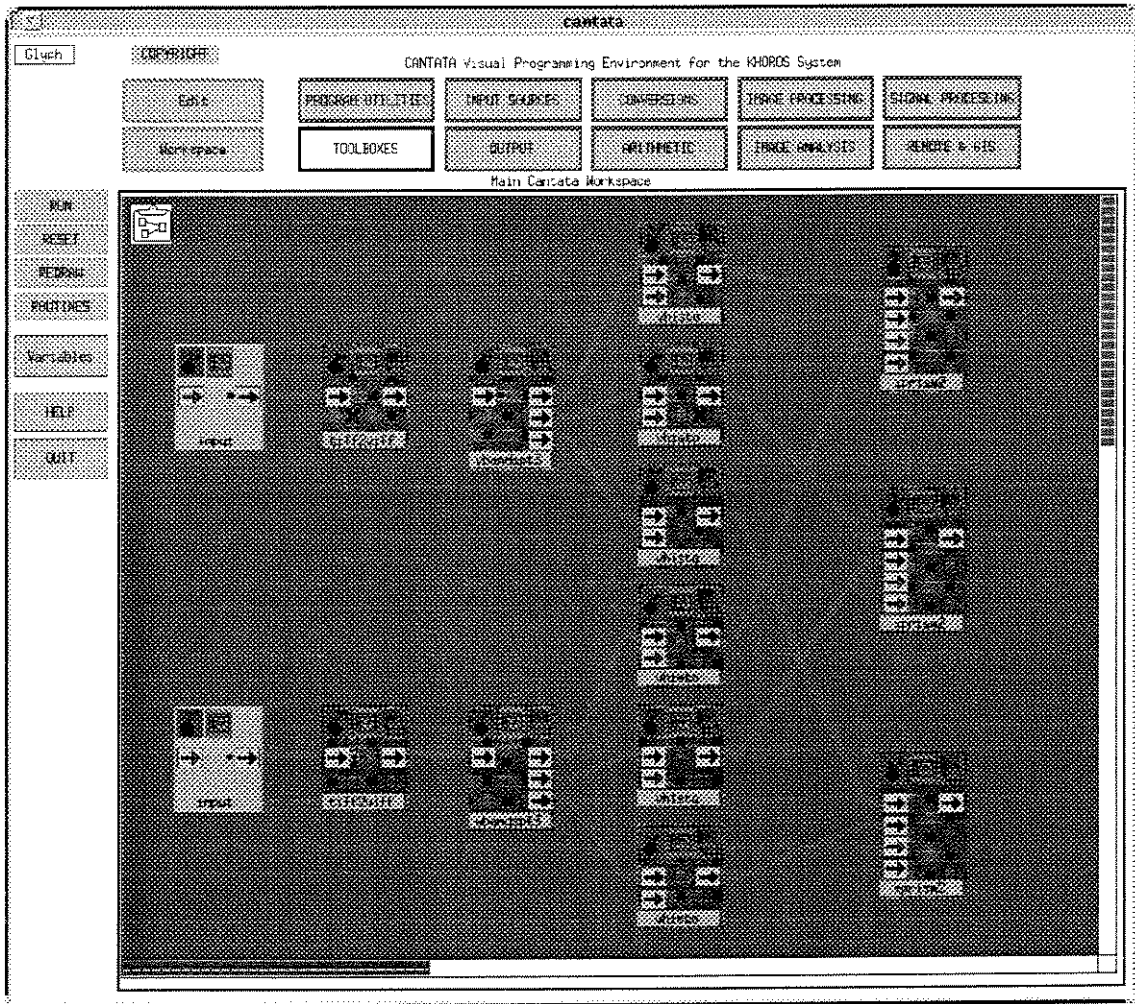


Fig. 5.16: "Workspace" do KHOROS utilizado para a obtenção dos histogramas.

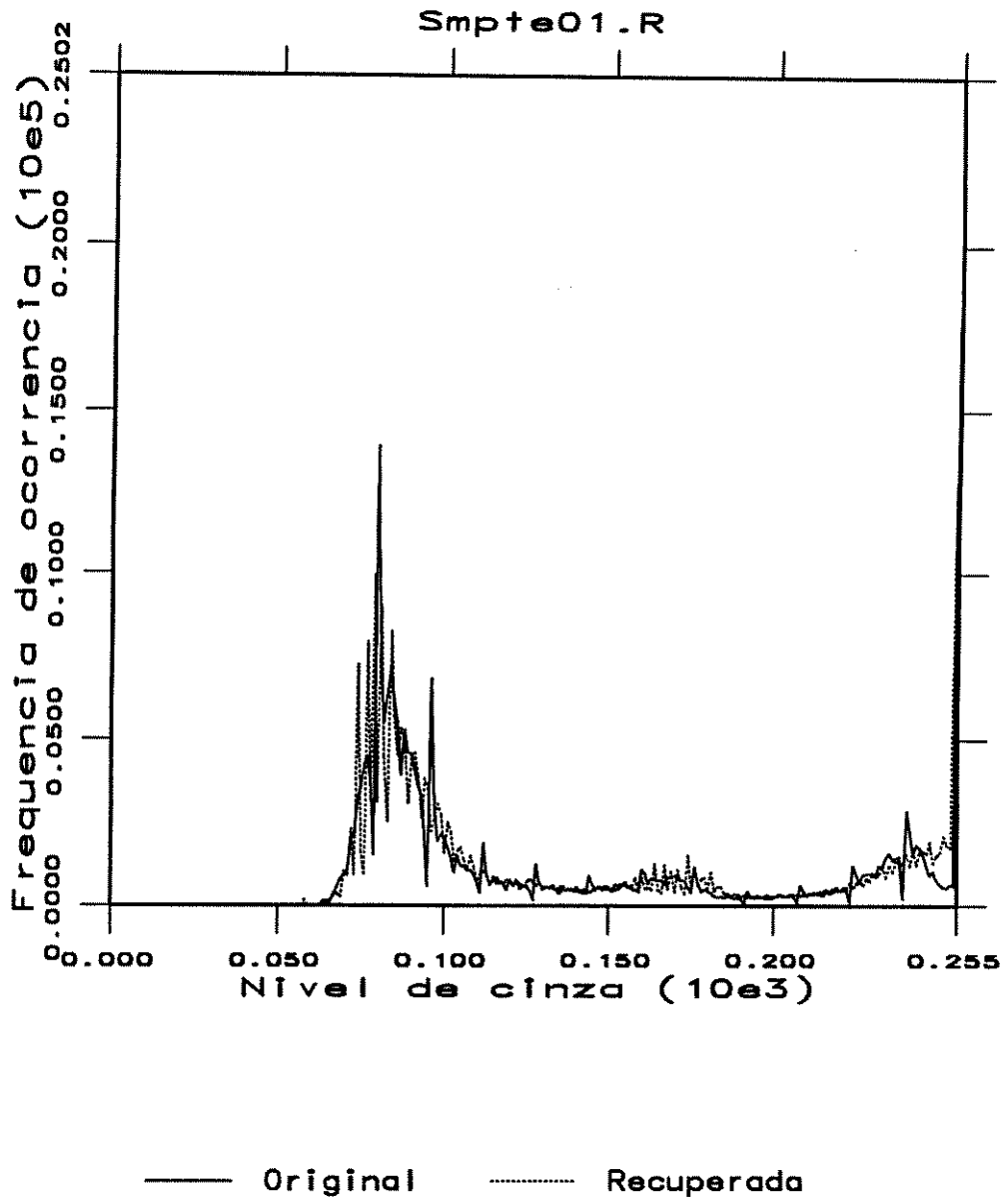


Fig 5.17: Histograma da componente R da imagem smpte01.

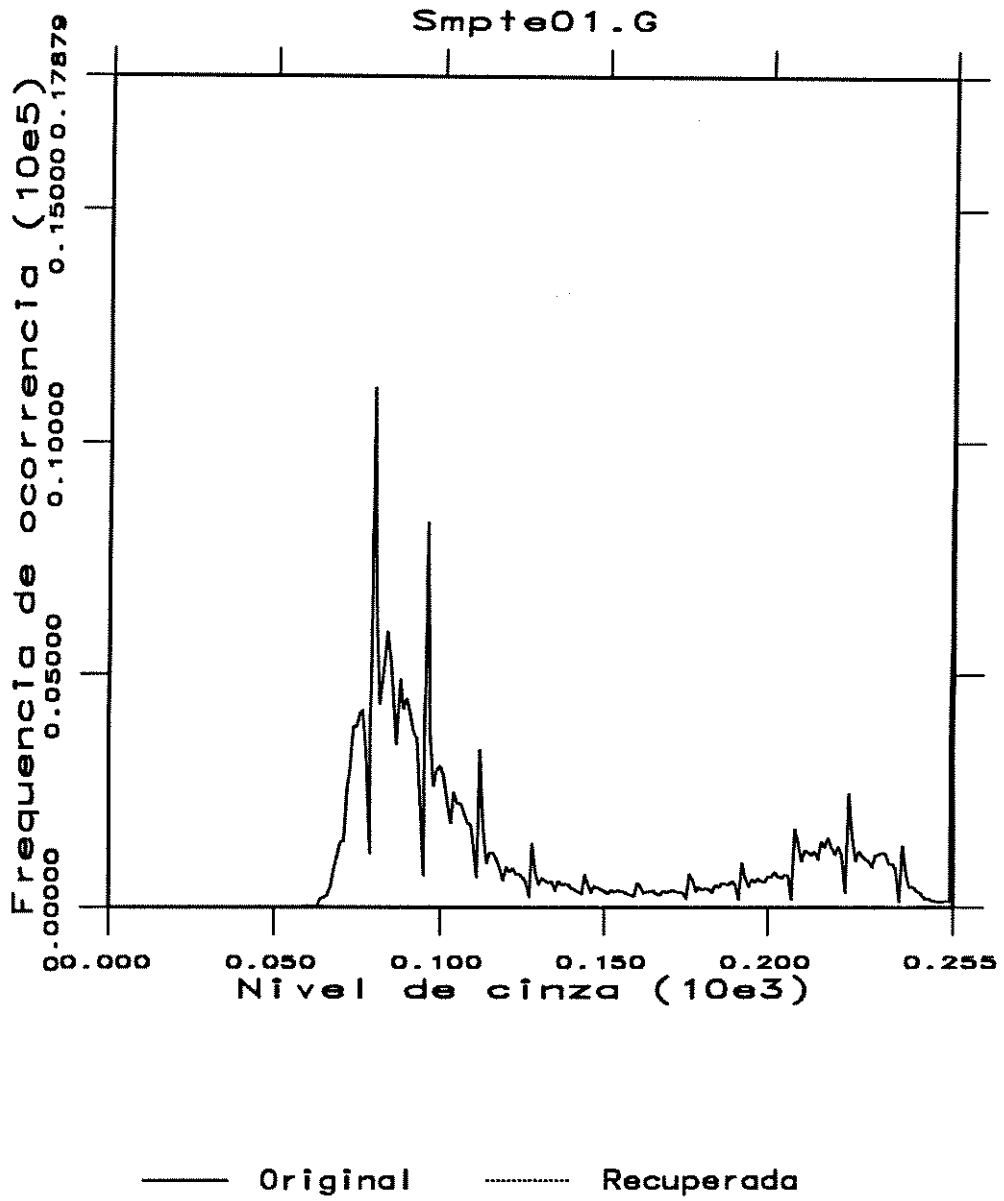


Fig 5.18: Histograma da componente G da imagem smpte01.

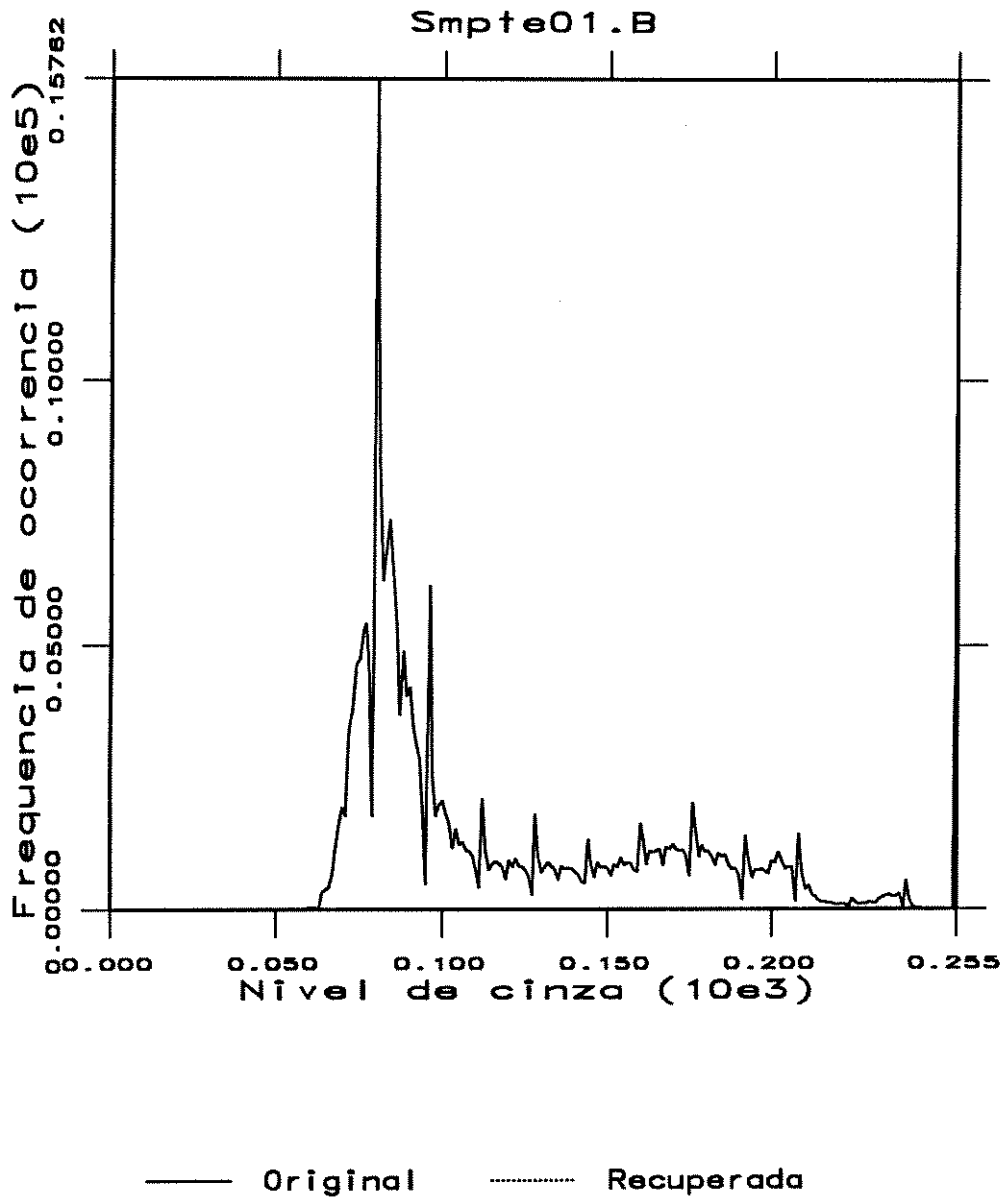


Fig 5.19: Histograma da componente B da imagem smpte01.

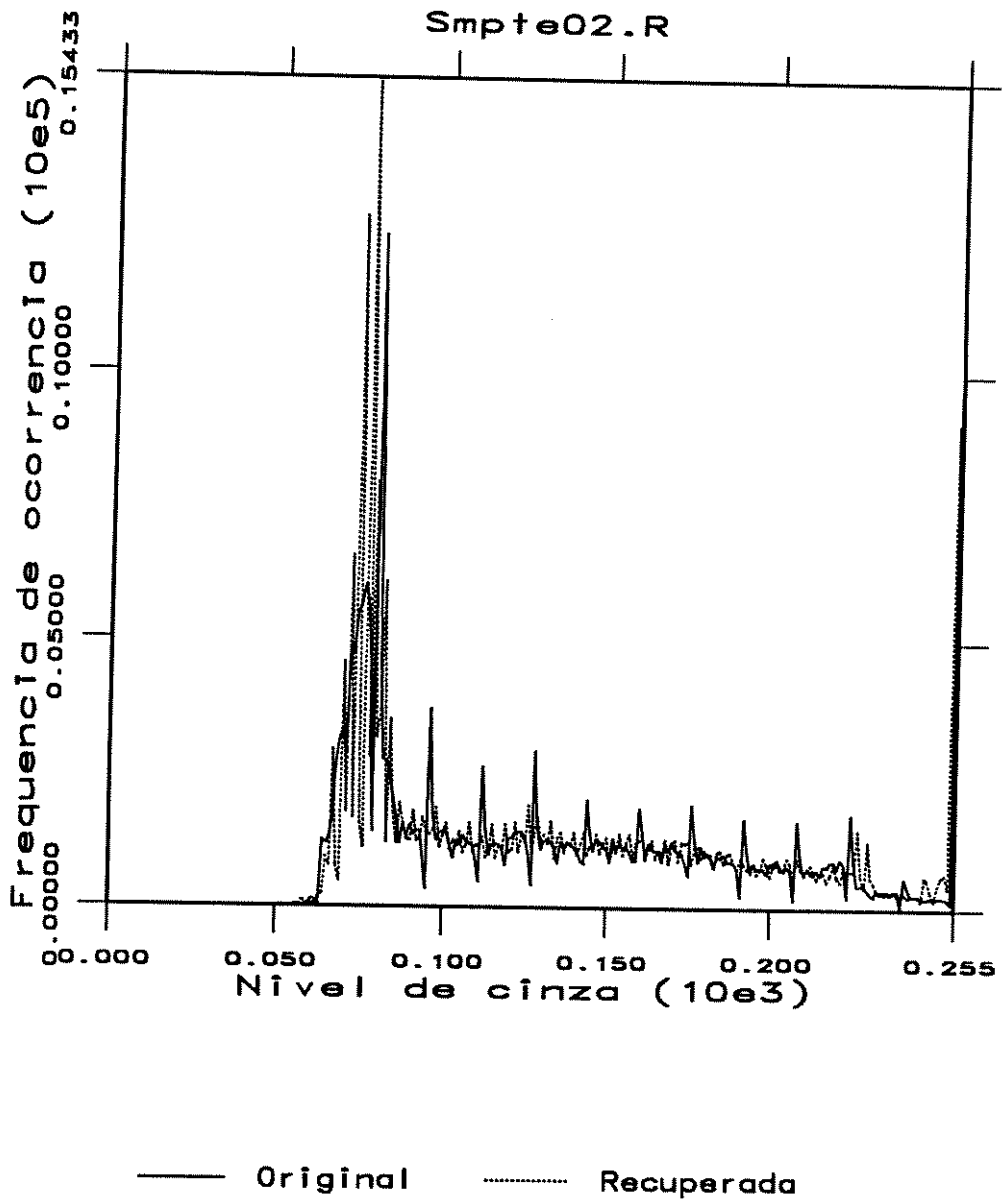


Fig 5.20: Histograma da componente R da imagem smpte02.

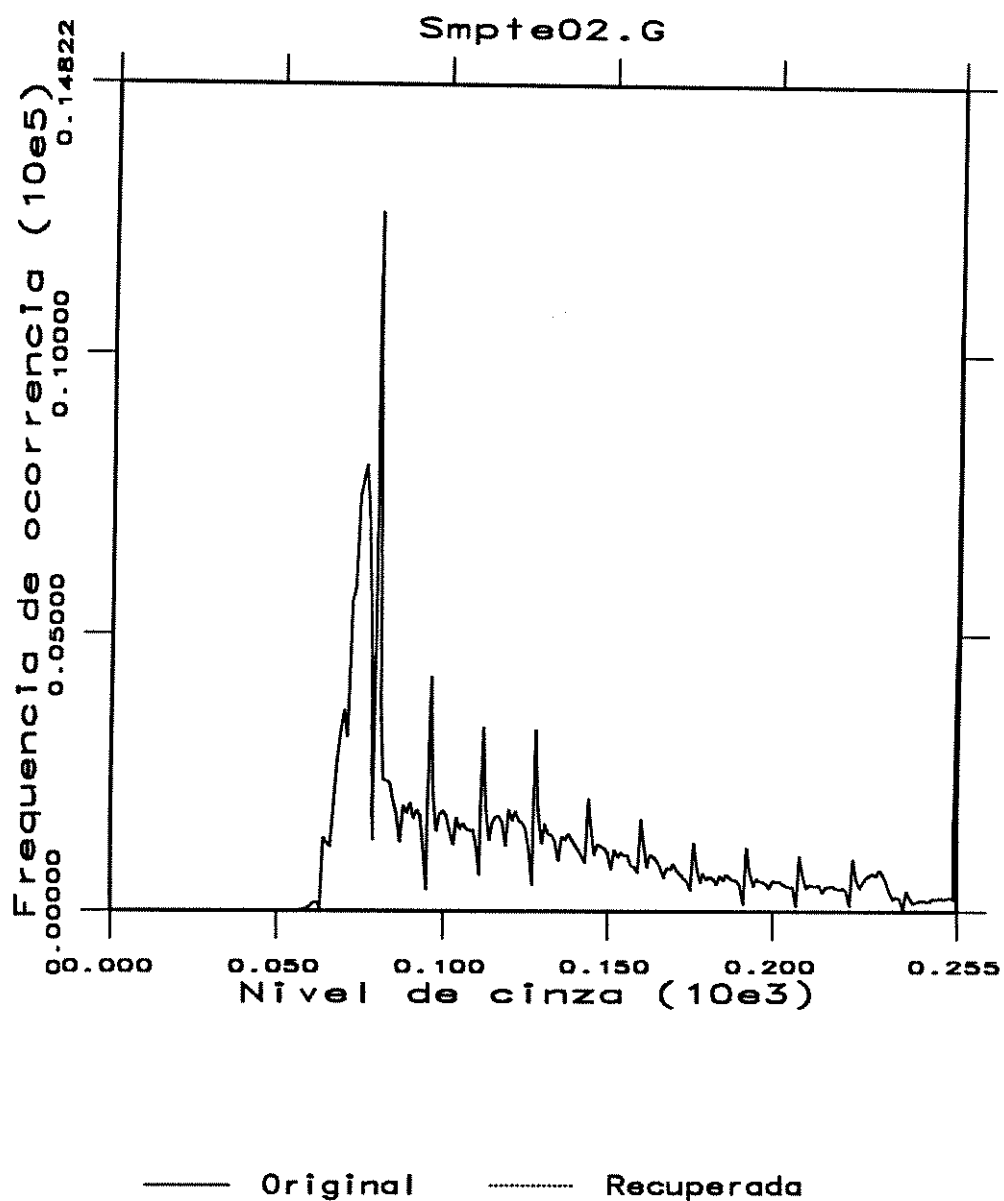


Fig 5.21: Histograma da componente G da imagem smpte02.

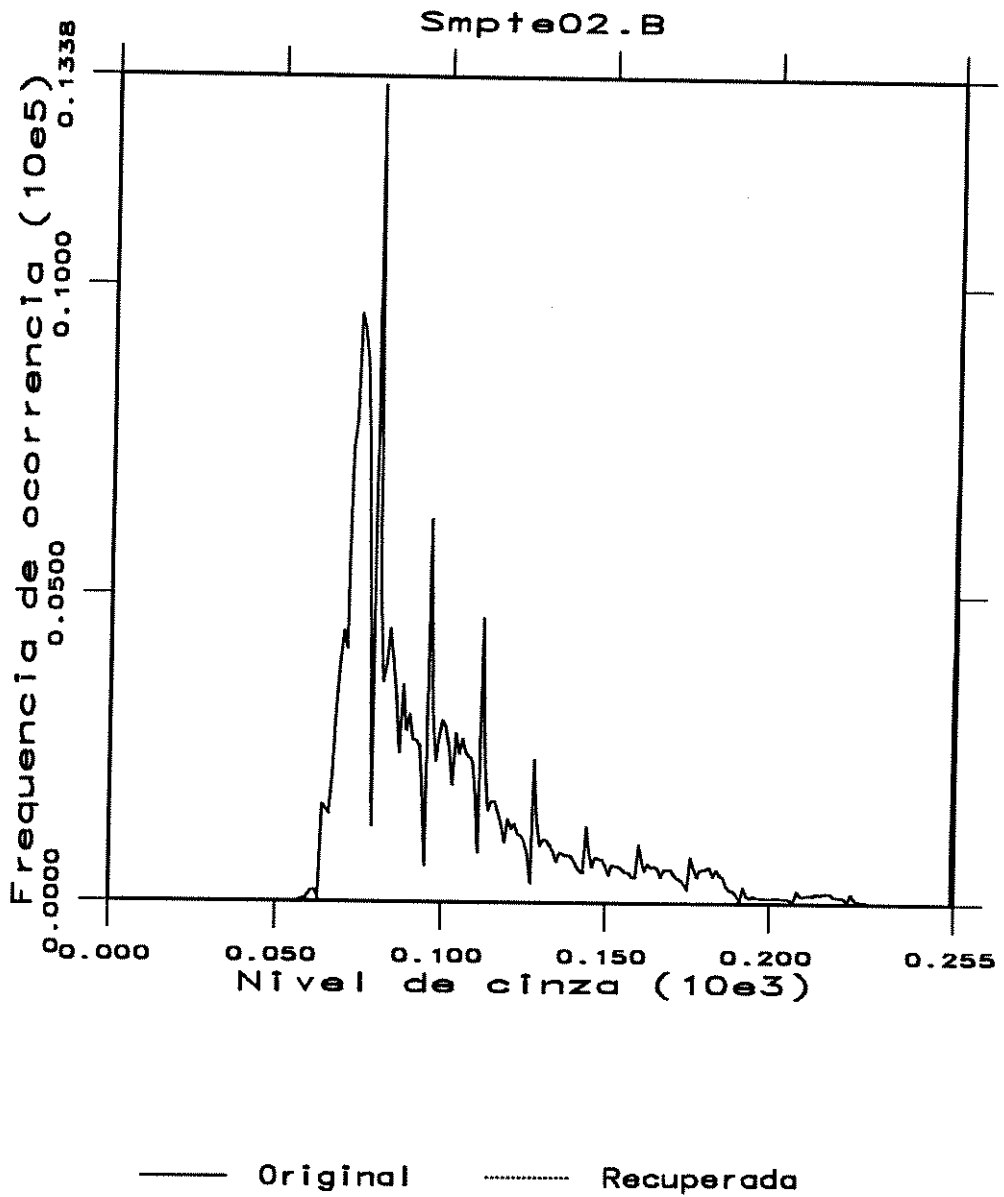


Fig 5.22: Histograma da componente B da imagem smpte02.

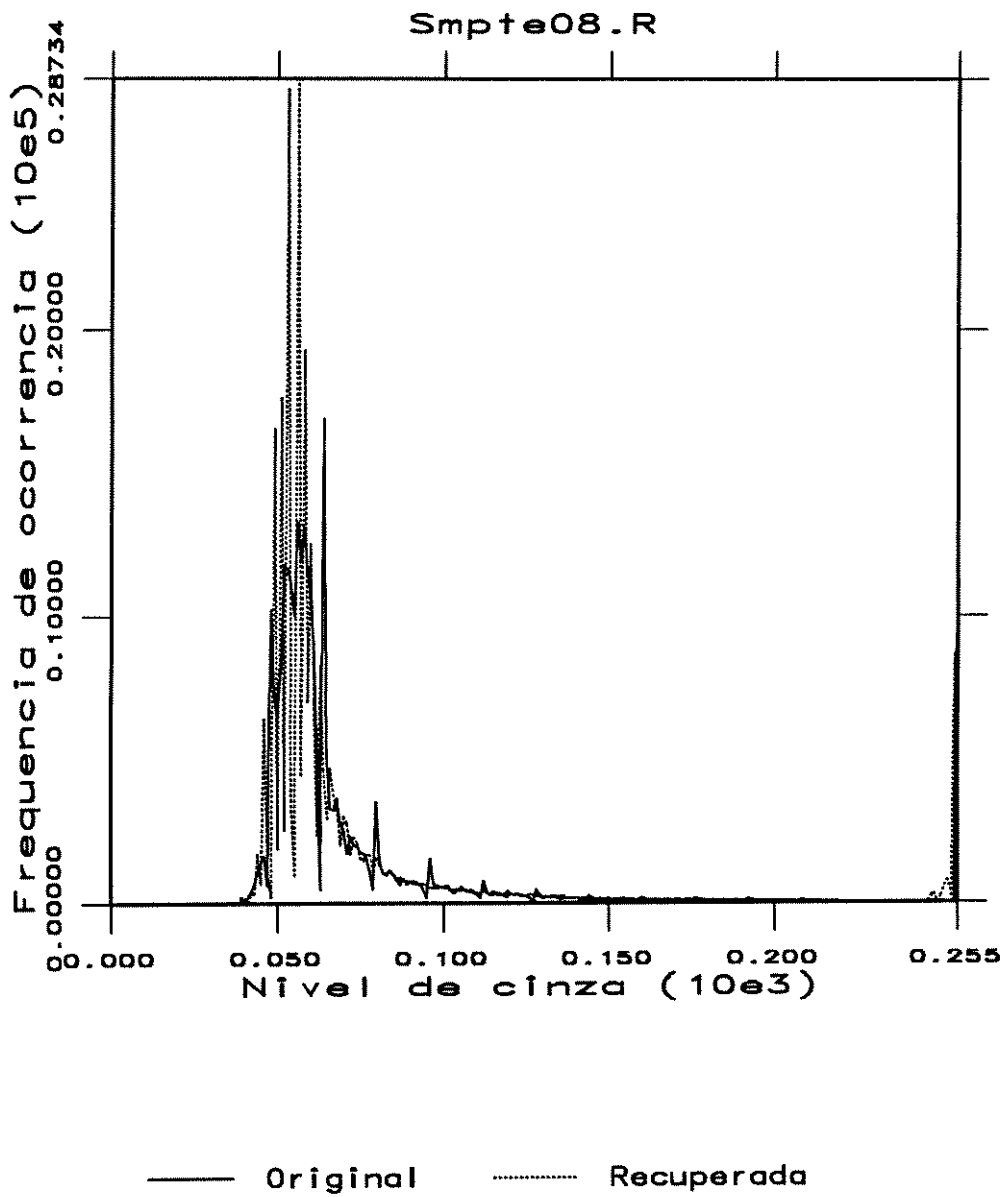


Fig 5.23: Histograma da componente R da imagem smpte08.

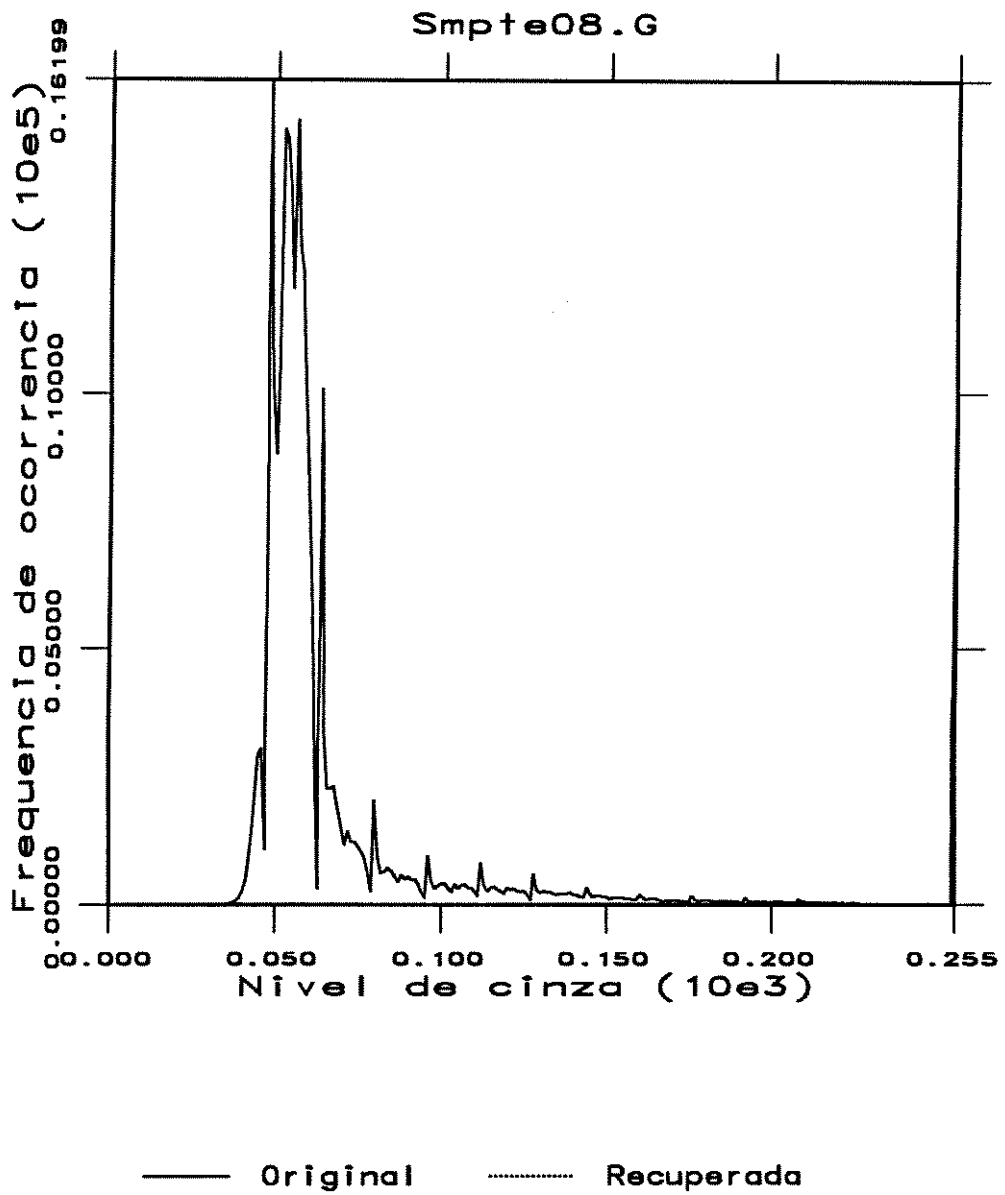


Fig 5.24: Histograma da componente G da imagem smpte08.

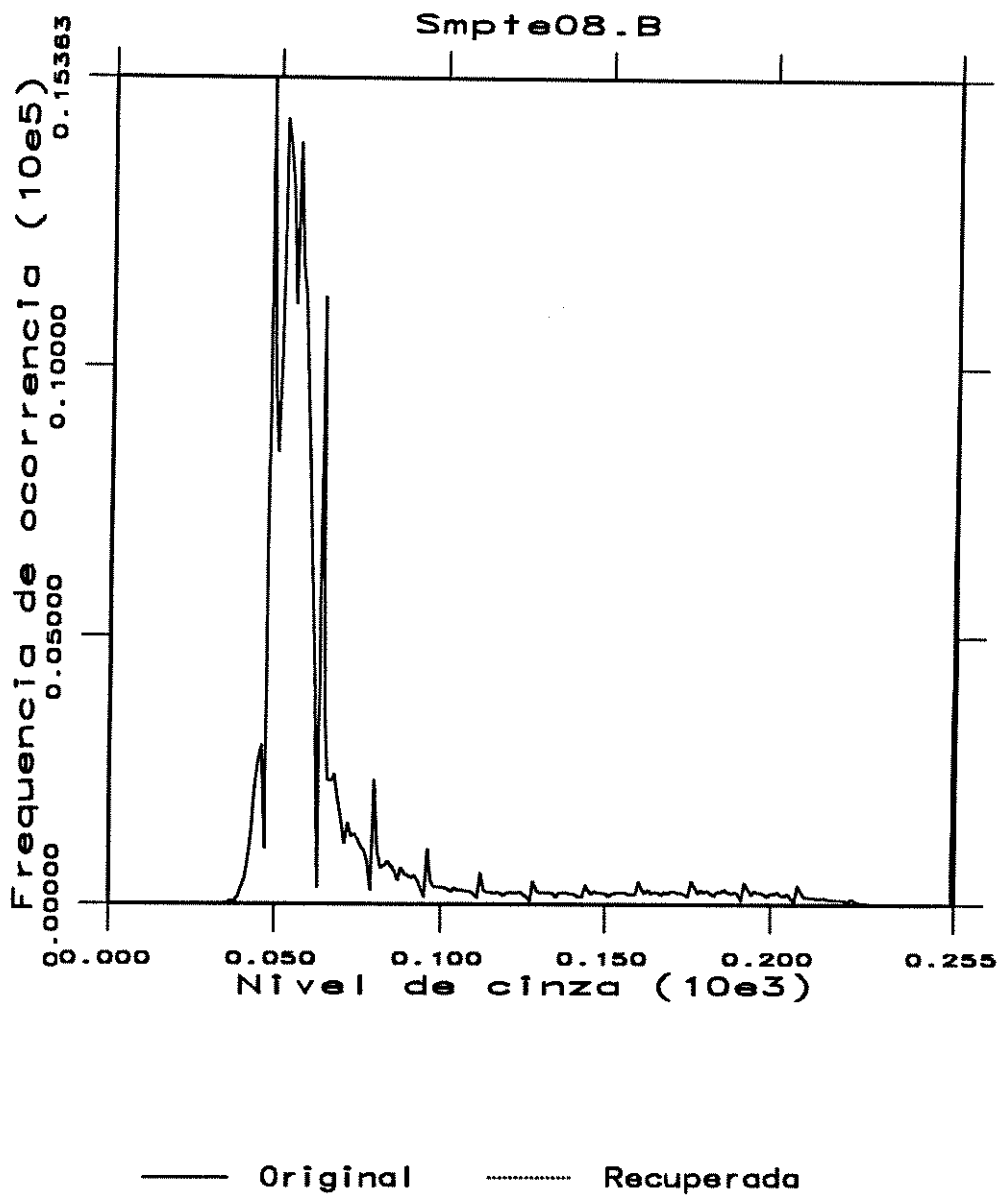


Fig 5.25: Histograma da componente B da imagem smpte08.

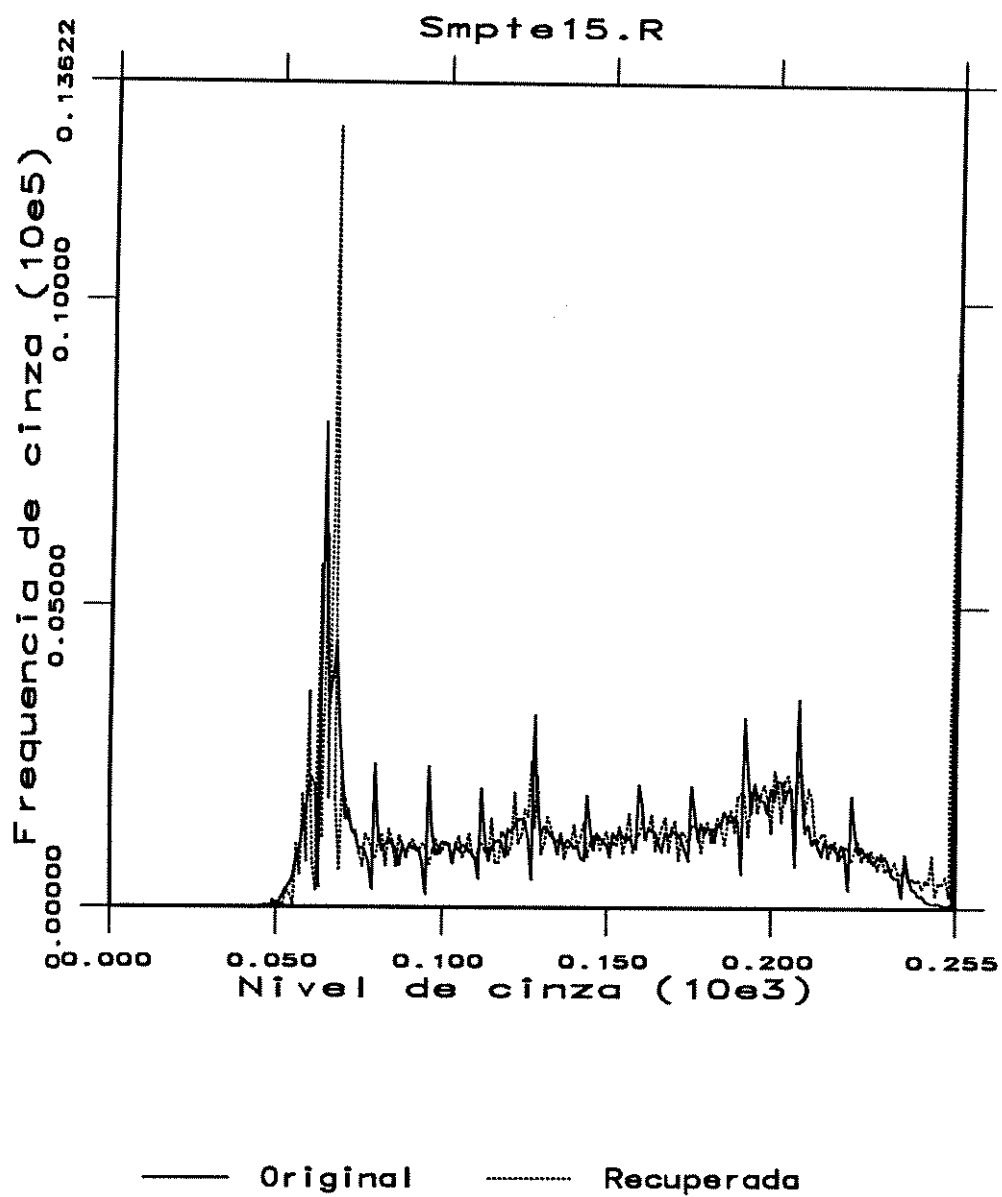


Fig 5.26: Histograma da componente R da imagem smpte15.

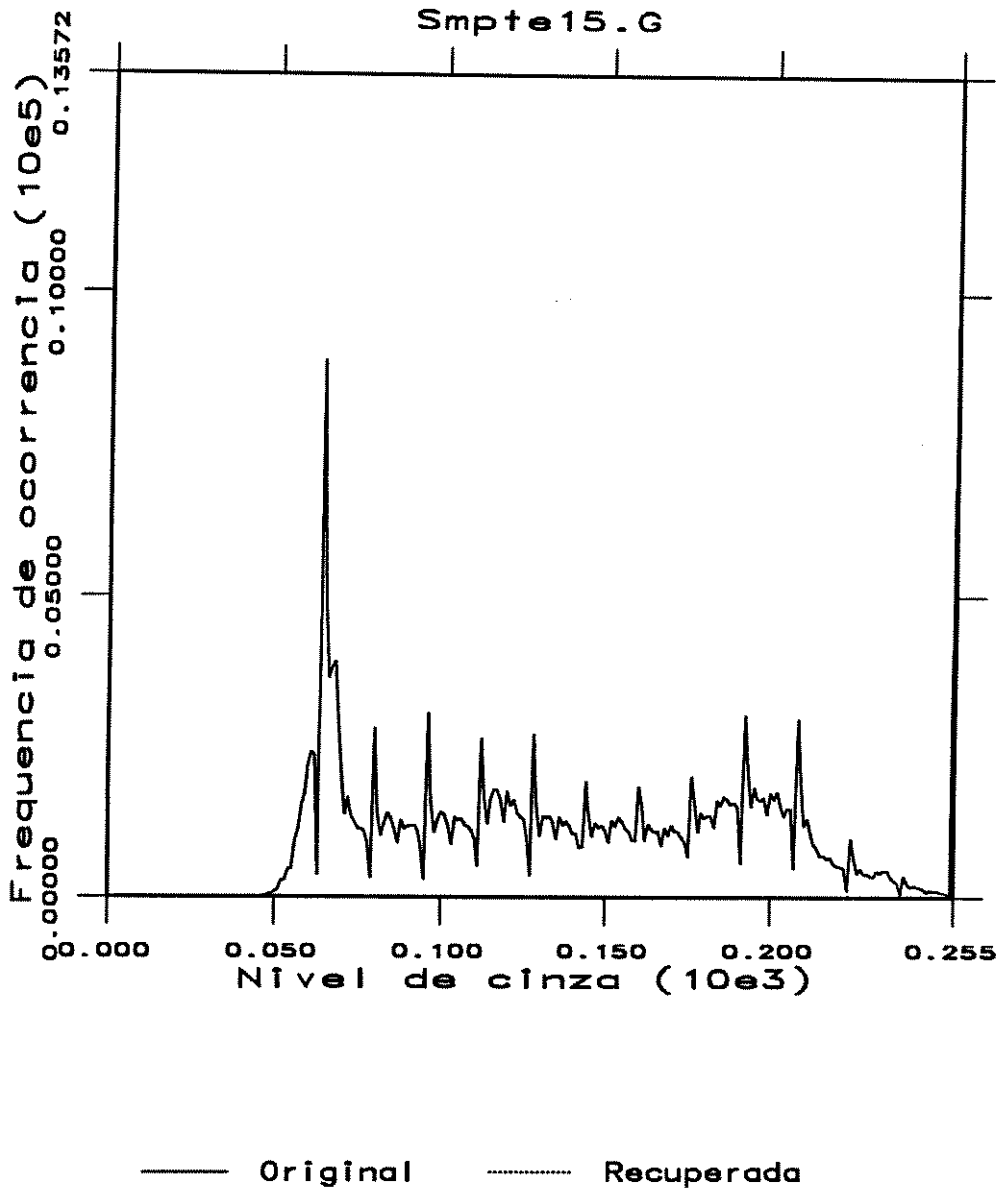


Fig 5.27: Histograma da componente G da imagem smpte15.

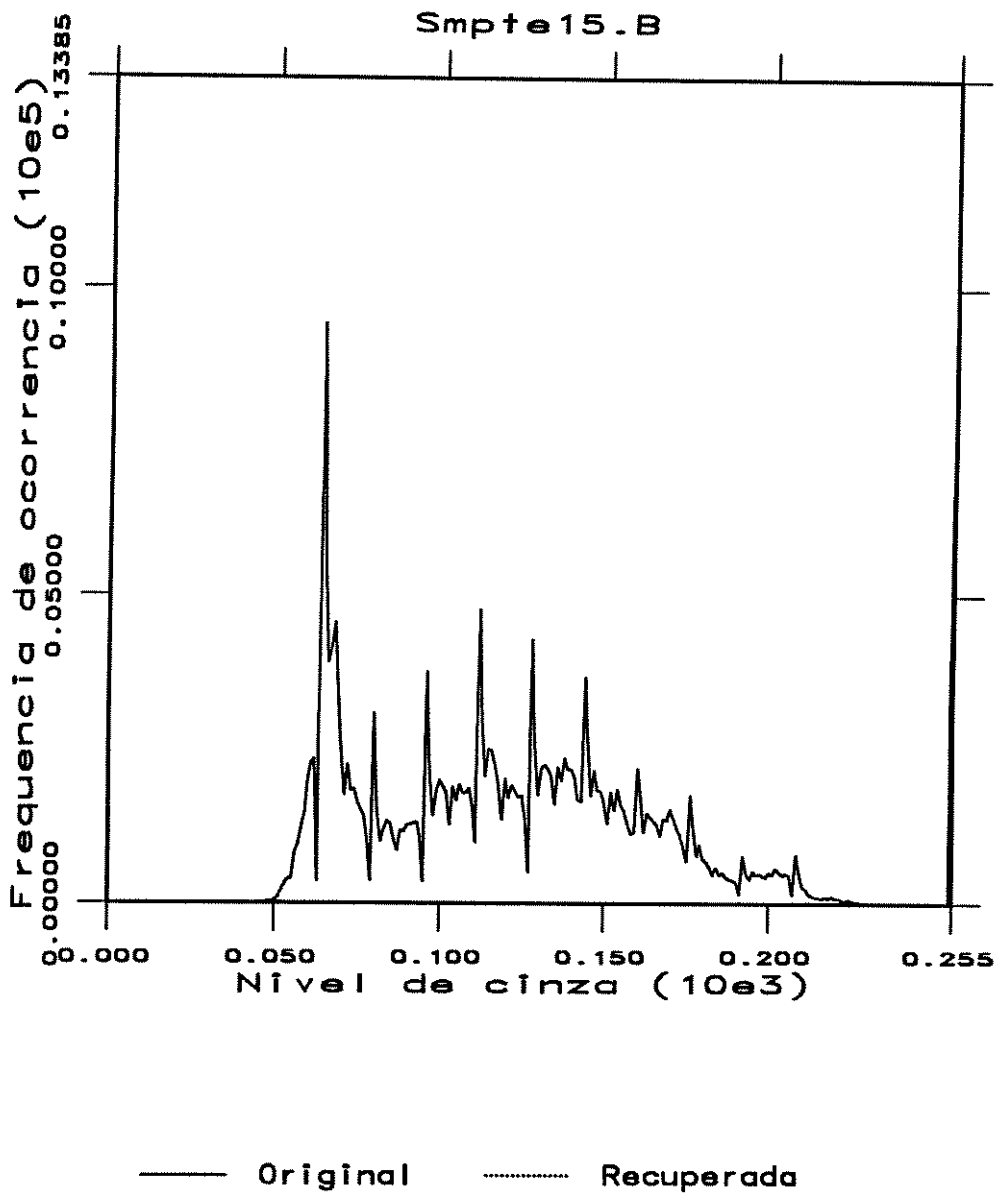


Fig 5.28: Histograma da componente B da imagem smpte15.

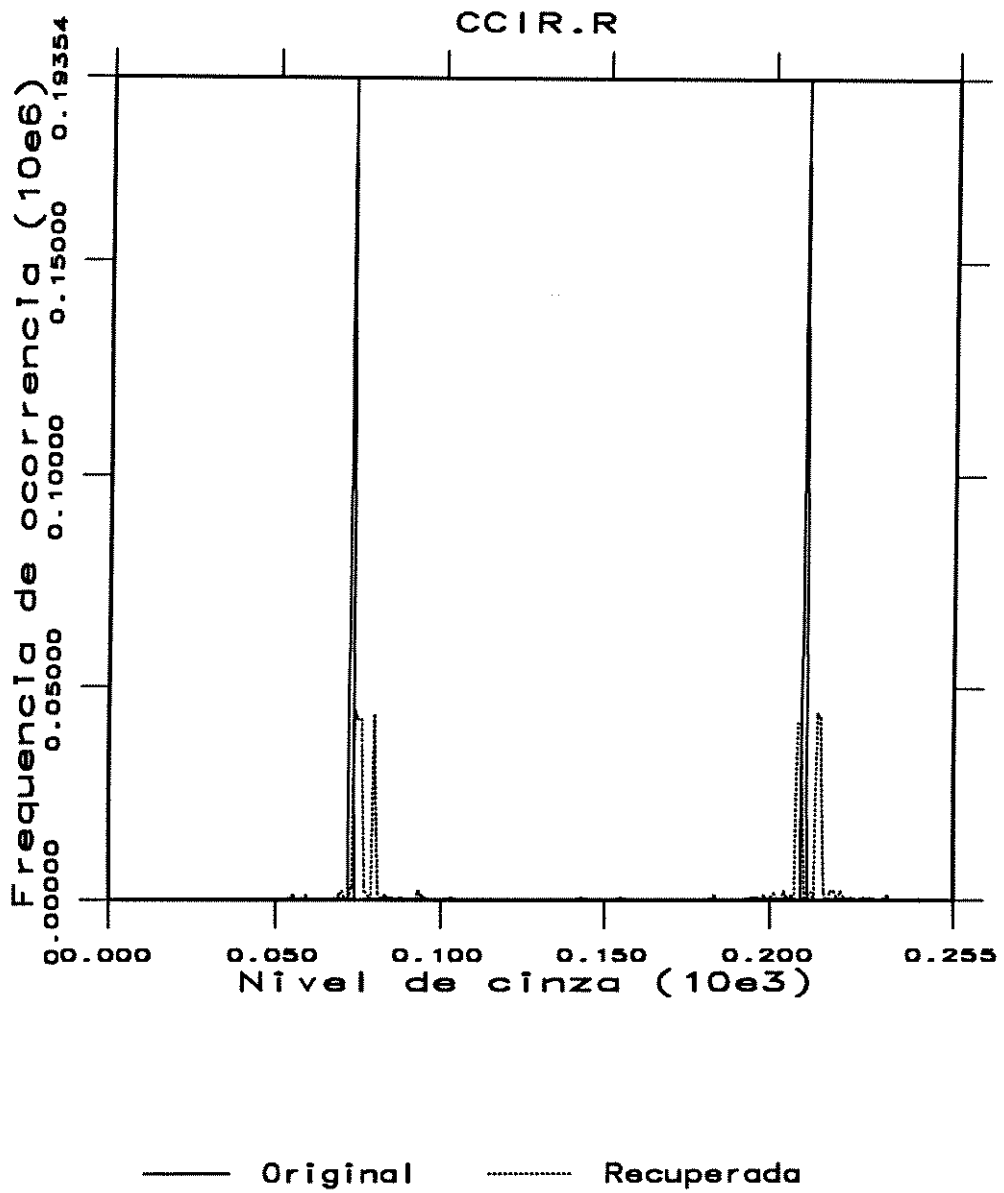


Fig 5.29: Histograma da componente R da imagem barras coloridas.

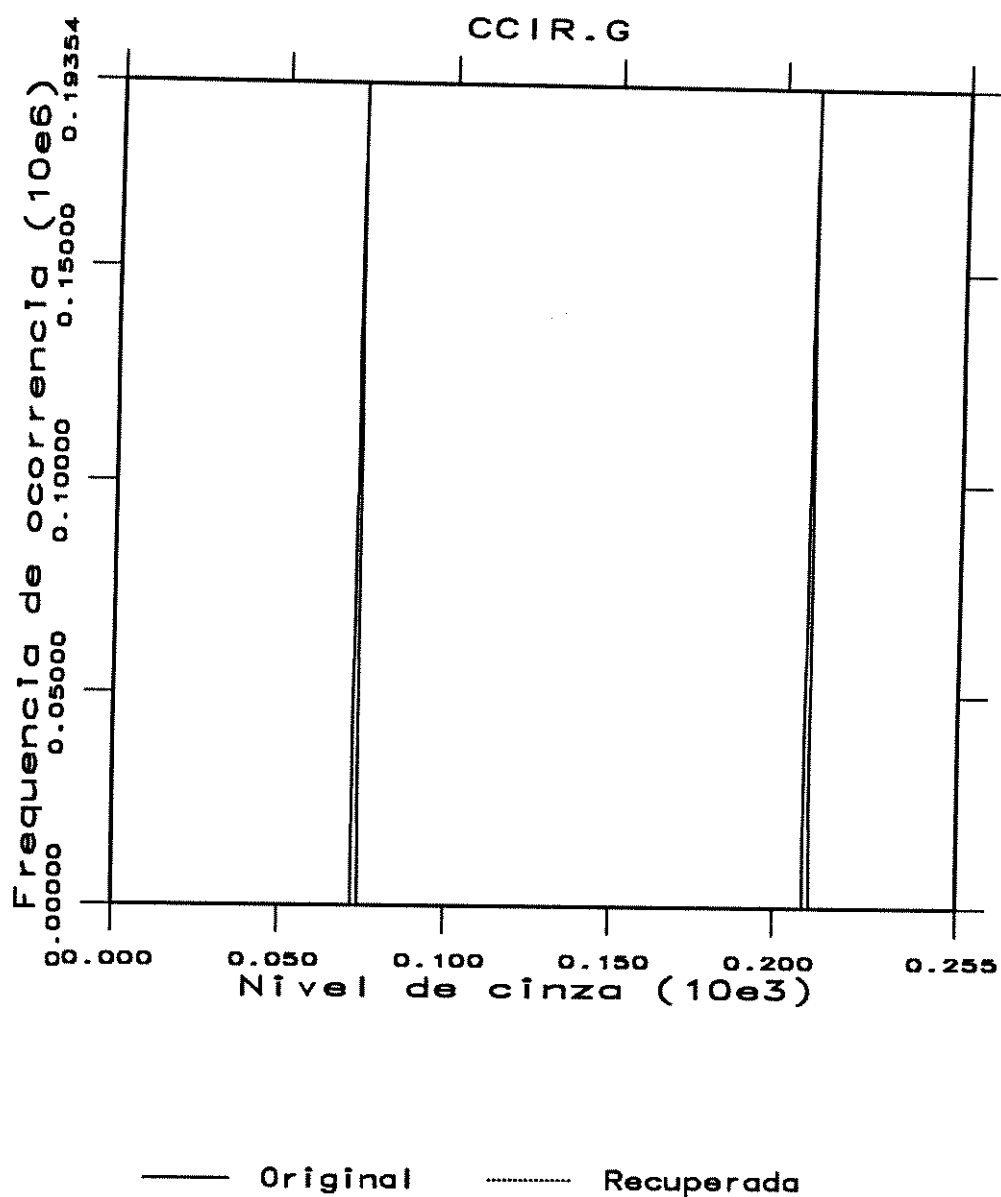


Fig 5.30: Histograma da componente G da imagem barras coloridas.

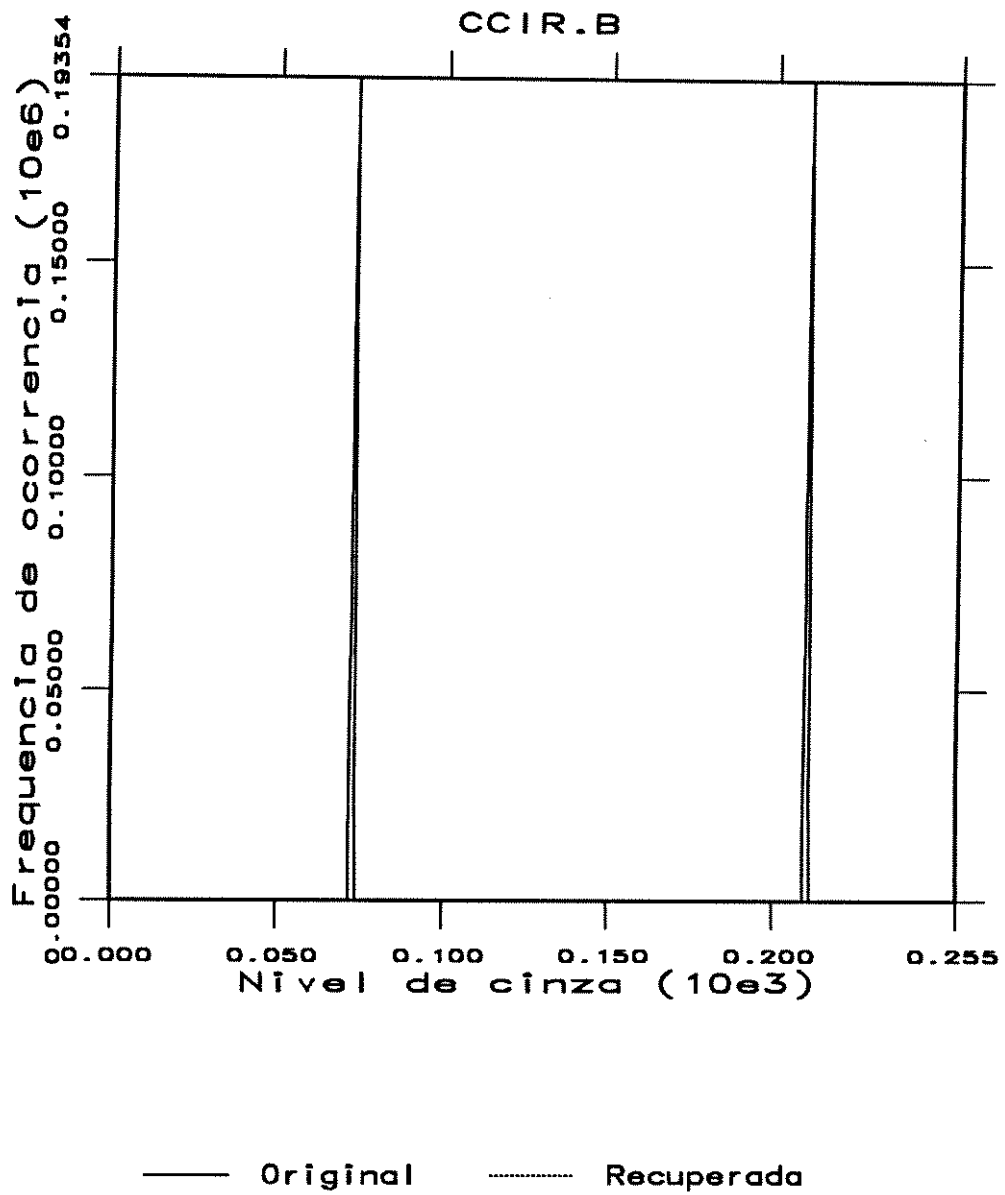


Fig 5.31: Histograma da componente B da imagem barras coloridas.

Pode-se observar nos histogramas das componentes das imagens, que as intensidades médias dos pixels distribuídos nas imagens recuperadas fazem um bom acompanhamento das originais, o que significa que as imagens processadas pelo sistema não sofreram mudança forte na intensidade dos pixels, permitindo- nos reafirmar com maior segurança, a alta fidelidade alcançada.

5.4.6 Ferramentas Utilizadas

As ferramentas computacionais usadas para o pré-processamento, processamento, pós-processamento e avaliações tanto subjetiva quanto objetiva do sistema simulado, são listadas a seguir. Cabe ressaltar que muitas delas encontram-se disponíveis na rede.

Conversores de Formato

Programas disponíveis para a conversão entre diferentes formatos de imagem (todos eles estão disponíveis na rede do DECOM).

- `rgbtogm`: programa de conversão do formato DAT de um arquivo de componente (R, G ou B) no formato PGM.
- `rawtoppm`: programa de conversão do formato de um conjunto de três arquivos de componentes (RGB) ao formato PPM.
- `ppmtoyuv`: programa de conversão do formato PPM de um arquivo de imagem ao formato Y, U e V. As crominâncias de entrada são subamostradas por um fator de dois nas dimensões horizontal e vertical.
- `yuvtoppm`: programa de conversão do formato YUV de três arquivos de componentes ao formato PPM.

Linguagem C

Para o desenvolvimento dos diversos programas de codificação e avaliação.

- jpeg.c: este arquivo contém as rotinas principais dos blocos do sistema JPEG básico.
- trans.c: este arquivo contém as rotinas dos blocos da DCT, quantização e zig-zag.
- dct.c (Department of Mathematics, University Stanford, 1992): este arquivo realiza a DCT pela fatoração matricial segundo Chen, Smith e Fraclik.
- cod.c: este arquivo contém as rotinas que manipulam a codificação e decodificação das cadeias de Huffman.
- huff.c: este arquivo contém as rotinas de codificação e decodificação de Huffman.
- control.c: este arquivo manipula os arquivos de imagem.
- mse.c: este arquivo determina o erro quadrático médio.
- snr.c: este arquivo determina a relação sinal/ruído.

Utilitários

Para o tratamento de imagens e construção de janelas de controle.

- XV Version 3.00 (John Bradley, University of Pennsylvania, 1993): para a visualização das imagens e conversão entre alguns formatos.
- KHOROS (The Khoros Group, Department of Electrical and Computer Engineering, University of New Mexico, Albuquerque): para a construção de um “workspace” para visualização de imagens e histogramas.
- OPEN WINDOWS DEVELOPER’S GUIDE 1.1: projetada para facilitar o desenvolvimento de interfaces dentro do Open Windows..

Capítulo 6

Conclusões e Sugestões

Neste trabalho foram investigados os aspectos mais relevantes do codec JPEG básico para a codificação e decodificação de imagens paradas de tons contínuos. Como foi estudado, esse codec é resultado da combinação de várias técnicas eficientes de compressão, tais como a codificação por transformada cosseno, modulação por código de pulso, codificação run-length e codificação de Huffman.

Deve-se ressaltar que esse trabalho não teve a pretensão de criar algo novo, mas sim, colocar uma base eficiente de partida para futuros estudos, adaptações e otimizações. Além disso, visa-se sua utilização na codificação de uma seqüência de imagens (imagens em movimento).

Assim avaliou-se o desempenho do sistema por meio de simulações em computador, quanto à eficiência da extração de redundância dos sinais e, principalmente, quanto à qualidade da imagem recuperada. Essa eficiência foi medida objetivamente, através dos parâmetros de taxa de bits de transmissão e da relação sinal/ruído; e subjetivamente, através da visualização das mesmas e análise dos respectivos histogramas.

Foi observado, que a subamostragem efetiva das componentes de crominância não provocaram mudança alguma na imagem recuperada, demonstrando o que já se tinha mencionado de que a maioria das componentes de alta freqüência (detalhes) concentram-se na luminância. Também, de acordo com os resultados determinados, a taxa de compressão depende do conteúdo de detalhes da imagem e que o sinal determinístico apresenta maior facilidade de redução da taxa do que as imagens da SMPTE. Esse resultado já era esperado, dado que os sinais determinísticos não possuem variações na vertical.

Os resultados das simulações foram satisfatórios, uma vez que a redução da redundância foi alta, a taxa de bits estava dentro da largura de faixa do canal de transmissão de 34 Mbit/s e as imagens recuperadas apresentaram elevada fidelidade visual com respeito à imagem original.

Conclui-se então que a plataforma de simulação do codec JPEG básico implementada é eficiente e pode servir de base para futuros estudos na codificação de imagens paradas.

Sugere-se para futuros trabalhos testar novas técnicas de codificação por transformada (Seno, Hadamard, Haar, Slant, Ondeletas, etc.). Na etapa da varredura “zig-zag” pode-se fazer uso de outras formas de varreduras (vertical, horizontal, mista, etc). Na codificação de Huffman, sugere-se criar os códigos de Huffman a partir da estatística do próprio sinal de entrada. Mudanças na etapa da quantização, como a utilização de técnicas de quantização adaptativas podem proporcionar melhores resultados, mas com um aumento da complexidade do sistema. Pode-se implementar os outros modos de operação do padrão JPEG (estendido, sem perda e hierárquico). Finalmente, deve-se usar a experiência adquirida na implementação da plataforma do JPEG base como ponto de partida para a codificação de uma seqüências de imagens em movimento, como propõe o MPEG (Moving Picture Experts Group).

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] HAYKIN, S. "Digital Communications.", John Wiley and Sons, New York, 1988.

- [2] ZOU, W. Y. " Digital HDTV Compression Techniques for Terrestrial Broadcasting.", SMPTE Journal, pp 127-131, Fevereiro 1993.

- [3] LIM, J. S. "Two Dimensional Signal and Image processing.", Prentice Hall, 1990.

- [4] JAIN, A. K. "Image Data Compression: A Review.", Proc. of the IEEE, vol. 69, 3, 1981.

- [5] KONSTATINOS, K. e BHASKARAN, V. "Imagem Compression Standards and Architectures.", ICASSP, 1994.

- [6] SHANNON, C. E. "The mathematical theory of communication." Partes I e II, Bell Syst. Tech. J., vol.27, pp.379 e 623, 1948.

- [7] GALLAGER, R. G. "Information Theory and Reliable Communication", John Wiley and Sons, New York, 1968.

- [8] HUFFMAN, D. A. "A method for the construction of minimum redundancy codes." Proceedings IRE, vol.40, pp.1098-1101, 1962.

- [9] NETRAVALI, A. N. e HASKELL, B. G. "Digital Pictures.", Plenum Press, New York, 1988.

- [10] JAIN, A. K. "Fundamentals of Digital Image Processing", Prentice Hall, 1989.
- [11] JAYANT, N. S. e NOLL, P. "Digital Coding of Waveforms", Prentice Hall, New Jersey, 1984.
- [12] OPPENHEIM, A. V. e SCHAFER, R. W. "Discrete- Time Signal Processing" Prentice Hall, New Jersey, 1989.
- [13] KITAJIMA, H. "A symmetric cosine transform." IEEE Trans. Comput., vol.C-29, pp.317-323, Abril 1980.
- [14] WANG, Z. "Fast algorithms for the discrete W transform and for the discrete Fourier transform." IEEE Trans. Acoust., Speech, and Signal Process, vol.ASSP-32, pp.803-816, Agosto 1984.
- [15] AHMED, N., NATARAJAN, T. e RAO, K. R. "Discrete Cosine Transform." IEEE Trans. Comput., vol.C-23, pp.90-93, Janeiro 1974.
- [16] RAO, K. R. e YIP, P. "Discrete Cosine Transform." Academic Press, 1990.
- [17] DAVENPORT Jr., W. B. e ROOT, W. L. "An introduction to the theory of random signals and noise." McGraw-Hill, New York, 1958.
- [18] RAY, W. D. e DRIVER, R. M. "Further decomposition of the Karhunen-Loeve series representation of a stationary random process." IEEE Trans. Inform. Theory, vol.IT-16, pp.663-668, Novembro 1970.
- [19] JAIN, A. K. "A fast Karhunen-Loeve transform for a class of random process." IEEE Trans. Commun., vol.COM-24, pp.1023-1029, Setembro 1976.

- [20] CHEN, W. H., SMITH, C. H. e FRALICK, S. C. "A fast computacional algorithm for the discrete cosine transform." IEEE Trans. Commun., vol.COM-25, pp.1004-1009, Setembro 1977.
- [21] WANG, Z. "Reconsideration of a fast computacional algorithm for the discrete cosine transform." IEEE Trans. Commun., vol.COM-31, pp.121-123, Janeiro 1983.
- [22] CHEN, T. C., GOTTLIEB, A. e SUN, M. T. "VLSI implementation of a 16x16 DCT." ICASSP 88, Intl. Conf. Acoust., Speech, and Signal Process, pp.1973-1976, New York, Abril 11-14, 1988.
- [23] WALLACE, G. K. "The JPEG still picture compression standard." Commun. of the ACM, vol.34, pp.31-44, Abril 1991.
- [24] HUDSON, G. P., YASUDA, H. e SEBESTYEN, I. "The International Standardisation of a Still Picture Compression Technique.", GLOBECOM '88, JPEG-249, 1988.
- [25] JPEG, "Information Technology. Digital Compression and Coding of continuous-tone still images.", ISO/IEC IS 10918-1, 1992.
- [26] CCIR, "Recommendation and Reports of the CCIR.", Broadcasting Service (Television), vol. XI, 1978.
- [27] SCHILDT, H. "C Completo e Total.", McGraw Hill, 1990.
- [28] DONOHOE, G. W. "Image Processing Short Course with KHOROS.", University of New Mexico, 1992.

ANEXO

A. Tabelas de Huffman Típicas para a Luminância e Crominância com 8 bits de Precisão.

- Tabelas de Huffman típicas para as diferenças de coeficientes DC

Os quadros A.1 e A.2 apresentam as tabelas de Huffman das diferenças de coeficientes desenvolvidas a partir de dados estatísticos médios tomando-se uma quantidade grande de imagens de vídeo com precisão de 8 bits. O quadro A.1 corresponde às componentes de luminância e o A.2 as de crominância. Embora não existam tabelas “default”, essas tabelas sugeridas podem ser úteis em muitas aplicações.

Quadro A.1: Tabela das diferenças de coeficientes DC de luminância.

| Category | Code length | Code word |
|----------|-------------|-----------|
| 0 | 2 | 00 |
| 1 | 3 | 010 |
| 2 | 3 | 011 |
| 3 | 3 | 100 |
| 4 | 3 | 101 |
| 5 | 3 | 110 |
| 6 | 4 | 1110 |
| 7 | 5 | 11110 |
| 8 | 6 | 111110 |
| 9 | 7 | 1111110 |
| 10 | 8 | 11111110 |
| 11 | 9 | 111111110 |

Quadro A.2: Tabela das diferenças de coeficientes DC de croma.

| Category | Code length | Code word |
|----------|-------------|-------------|
| 0 | 2 | 00 |
| 1 | 2 | 01 |
| 2 | 2 | 10 |
| 3 | 3 | 110 |
| 4 | 4 | 1110 |
| 5 | 5 | 11110 |
| 6 | 6 | 111110 |
| 7 | 7 | 1111110 |
| 8 | 8 | 11111110 |
| 9 | 9 | 111111110 |
| 10 | 10 | 1111111110 |
| 11 | 11 | 11111111110 |

- **Tabelas Típicas para os coeficientes AC**

Os quadros A.3 e A.4 apresentam as tabelas de Huffman para os coeficientes AC desenvolvidos a partir de dados estatísticos médios de uma quantidade grande de imagens de vídeo com uma precisão de 8 bits. O quadro A.3 é o correspondente às componentes de luminância e o A.4 as de croma. Embora não existam tabelas com valores “default”, essas tabelas sugeridas podem ser úteis em muitas aplicações.

Quadro A.3: Tabela de coeficientes AC de luminância.

| Run/Size | Code length | Code word |
|-----------|-------------|-----------------|
| 0/0 (EOB) | 4 | 1010 |
| 0/1 | 2 | 00 |
| 0/2 | 2 | 01 |
| 0/3 | 3 | 100 |
| 0/4 | 4 | 1011 |
| 0/5 | 5 | 11010 |
| 0/6 | 7 | 1111000 |
| 0/7 | 8 | 11111000 |
| 0/8 | 10 | 1111110110 |
| 0/9 | 16 | 111111110000010 |
| 0/A | 16 | 111111110000011 |
| 1/1 | 4 | 1100 |
| 1/2 | 5 | 11011 |
| 1/3 | 7 | 1111001 |
| 1/4 | 9 | 111110110 |
| 1/5 | 11 | 11111110110 |
| 1/6 | 16 | 111111110000100 |
| 1/7 | 16 | 111111110000101 |
| 1/8 | 16 | 111111110000110 |
| 1/9 | 16 | 111111110000111 |
| 1/A | 16 | 111111110001000 |
| 2/1 | 5 | 11100 |
| 2/2 | 8 | 11111001 |
| 2/3 | 10 | 1111110111 |
| 2/4 | 12 | 111111110100 |
| 2/5 | 16 | 111111110001001 |
| 2/6 | 16 | 111111110001010 |
| 2/7 | 16 | 111111110001011 |
| 2/8 | 16 | 111111110001100 |
| 2/9 | 16 | 111111110001101 |
| 2/A | 16 | 111111110001110 |
| 3/1 | 6 | 111010 |
| 3/2 | 9 | 111110111 |
| 3/3 | 12 | 111111110101 |
| 3/4 | 16 | 111111110001111 |
| 3/5 | 16 | 111111110010000 |
| 3/6 | 16 | 111111110010001 |
| 3/7 | 16 | 111111110010010 |
| 3/8 | 16 | 111111110010011 |
| 3/9 | 16 | 111111110010100 |
| 3/A | 16 | 111111110010101 |

Quadro A.3: Tabela de coeficientes AC de luminância (cont.).

| Run/Size | Code length | Code word |
|----------|-------------|-----------------|
| 4/1 | 6 | 111011 |
| 4/2 | 10 | 111111000 |
| 4/3 | 16 | 111111110010110 |
| 4/4 | 16 | 111111110010111 |
| 4/5 | 16 | 111111110011000 |
| 4/6 | 16 | 111111110011001 |
| 4/7 | 16 | 111111110011010 |
| 4/8 | 16 | 111111110011011 |
| 4/9 | 16 | 111111110011100 |
| 4/A | 16 | 111111110011101 |
| 5/1 | 7 | 1111010 |
| 5/2 | 11 | 1111110111 |
| 5/3 | 16 | 111111110011110 |
| 5/4 | 16 | 111111110011111 |
| 5/5 | 16 | 111111110100000 |
| 5/6 | 16 | 111111110100001 |
| 5/7 | 16 | 111111110100010 |
| 5/8 | 16 | 111111110100011 |
| 5/9 | 16 | 111111110100100 |
| 5/A | 16 | 111111110100101 |
| 6/1 | 7 | 1111011 |
| 6/2 | 12 | 11111110110 |
| 6/3 | 16 | 111111110100110 |
| 6/4 | 16 | 111111110100111 |
| 6/5 | 16 | 111111110101000 |
| 6/6 | 16 | 111111110101001 |
| 6/7 | 16 | 111111110101010 |
| 6/8 | 16 | 111111110101011 |
| 6/9 | 16 | 111111110101100 |
| 6/A | 16 | 111111110101101 |
| 7/1 | 8 | 11111010 |
| 7/2 | 12 | 11111110111 |
| 7/3 | 16 | 111111110101110 |
| 7/4 | 16 | 111111110101111 |
| 7/5 | 16 | 111111110110000 |
| 7/6 | 16 | 111111110110001 |
| 7/7 | 16 | 111111110110010 |
| 7/8 | 16 | 111111110110011 |
| 7/9 | 16 | 111111110110100 |
| 7/A | 16 | 111111110110101 |
| 8/1 | 9 | 11111000 |
| 8/2 | 15 | 11111111000000 |

Quadro A.3: Tabela de coeficientes AC de luminância (cont.).

| Run/Size | Code length | Code word |
|----------|-------------|-----------------|
| 8/3 | 16 | 111111110110110 |
| 8/4 | 16 | 111111110110111 |
| 8/5 | 16 | 111111110111000 |
| 8/6 | 16 | 111111110111001 |
| 8/7 | 16 | 111111110111010 |
| 8/8 | 16 | 111111110111011 |
| 8/9 | 16 | 111111110111100 |
| 8/A | 16 | 111111110111101 |
| 9/1 | 9 | 111111001 |
| 9/2 | 16 | 111111110111110 |
| 9/3 | 16 | 111111110111111 |
| 9/4 | 16 | 111111111000000 |
| 9/5 | 16 | 111111111000001 |
| 9/6 | 16 | 111111111000010 |
| 9/7 | 16 | 111111111000011 |
| 9/8 | 16 | 111111111000100 |
| 9/9 | 16 | 111111111000101 |
| 9/A | 16 | 111111111000110 |
| A/1 | 9 | 111111010 |
| A/2 | 16 | 111111111000111 |
| A/3 | 16 | 111111111001000 |
| A/4 | 16 | 111111111001001 |
| A/5 | 16 | 111111111001010 |
| A/6 | 16 | 111111111001011 |
| A/7 | 16 | 111111111001100 |
| A/8 | 16 | 111111111001101 |
| A/9 | 16 | 111111111001110 |
| A/A | 16 | 111111111001111 |
| B/1 | 10 | 1111111001 |
| B/2 | 16 | 111111111010000 |
| B/3 | 16 | 111111111010001 |
| B/4 | 16 | 111111111010010 |
| B/5 | 16 | 111111111010011 |
| B/6 | 16 | 111111111010100 |
| B/7 | 16 | 111111111010101 |
| B/8 | 16 | 111111111010110 |
| B/9 | 16 | 111111111010111 |
| B/A | 16 | 111111111011000 |
| C/1 | 10 | 1111111010 |
| C/2 | 16 | 111111111011001 |
| C/3 | 16 | 111111111011010 |
| C/4 | 16 | 111111111011011 |

Quadro A.3: Tabela de coeficientes AC de luminância (cont.).

| Run/Size | Code length | Code word |
|-----------|-------------|-----------------|
| C/5 | 16 | 111111111011100 |
| C/6 | 16 | 111111111011101 |
| C/7 | 16 | 111111111011110 |
| C/8 | 16 | 111111111011111 |
| C/9 | 16 | 111111111100000 |
| C/A | 16 | 111111111100001 |
| D/1 | 11 | 1111111000 |
| D/2 | 16 | 111111111100010 |
| D/3 | 16 | 111111111100011 |
| D/4 | 16 | 111111111100100 |
| D/5 | 16 | 111111111100101 |
| D/6 | 16 | 111111111100110 |
| D/7 | 16 | 111111111100111 |
| D/8 | 16 | 111111111101000 |
| D/9 | 16 | 111111111101001 |
| D/A | 16 | 111111111101010 |
| E/1 | 16 | 111111111101011 |
| E/2 | 16 | 111111111101100 |
| E/3 | 16 | 111111111101101 |
| E/4 | 16 | 111111111101110 |
| E/5 | 16 | 111111111101111 |
| E/6 | 16 | 111111111110000 |
| E/7 | 16 | 111111111110001 |
| E/8 | 16 | 111111111110010 |
| E/9 | 16 | 111111111110011 |
| E/A | 16 | 111111111110100 |
| F/0 (ZRL) | 11 | 1111111001 |
| F/1 | 16 | 111111111110101 |
| F/2 | 16 | 111111111110110 |
| F/3 | 16 | 111111111110111 |
| F/4 | 16 | 111111111111000 |
| F/5 | 16 | 111111111111001 |
| F/6 | 16 | 111111111111010 |
| F/7 | 16 | 111111111111011 |
| F/8 | 16 | 111111111111100 |
| F/9 | 16 | 111111111111101 |
| F/A | 16 | 111111111111110 |

Quadro A.4: Tabela de coeficientes AC de crominância.

| Run/Size | Code length | Code word |
|-----------|-------------|------------------|
| 0/0 (EOB) | 2 | 00 |
| 0/1 | 2 | 01 |
| 0/2 | 3 | 100 |
| 0/3 | 4 | 1010 |
| 0/4 | 5 | 11000 |
| 0/5 | 5 | 11001 |
| 0/6 | 6 | 111000 |
| 0/7 | 7 | 1111000 |
| 0/8 | 9 | 111110100 |
| 0/9 | 10 | 1111110110 |
| 0/A | 12 | 111111110100 |
| 1/1 | 4 | 1011 |
| 1/2 | 6 | 111001 |
| 1/3 | 8 | 11110110 |
| 1/4 | 9 | 111110101 |
| 1/5 | 11 | 11111110110 |
| 1/6 | 12 | 111111110101 |
| 1/7 | 16 | 1111111110001000 |
| 1/8 | 16 | 1111111110001001 |
| 1/9 | 16 | 1111111110001010 |
| 1/A | 16 | 1111111110001011 |
| 2/1 | 5 | 11010 |
| 2/2 | 8 | 11110111 |
| 2/3 | 10 | 1111110111 |
| 2/4 | 12 | 111111110110 |
| 2/5 | 15 | 111111111000010 |
| 2/6 | 16 | 1111111110001100 |
| 2/7 | 16 | 1111111110001101 |
| 2/8 | 16 | 1111111110001110 |
| 2/9 | 16 | 1111111110001111 |
| 2/A | 16 | 1111111110010000 |
| 3/1 | 5 | 11011 |
| 3/2 | 8 | 11111000 |
| 3/3 | 10 | 1111111000 |
| 3/4 | 12 | 111111110111 |
| 3/5 | 16 | 1111111110010001 |
| 3/6 | 16 | 1111111110010010 |
| 3/7 | 16 | 1111111110010011 |
| 3/8 | 16 | 1111111110010100 |
| 3/9 | 16 | 1111111110010101 |
| 3/A | 16 | 1111111110010110 |
| 4/1 | 6 | 111010 |

Quadro A.4: Tabela de coeficientes AC de crominância (cont.).

| Run/Size | Code length | Code word |
|----------|-------------|-----------------|
| 4/2 | 9 | 111110110 |
| 4/3 | 16 | 111111110010111 |
| 4/4 | 16 | 111111110011000 |
| 4/5 | 16 | 111111110011001 |
| 4/6 | 16 | 111111110011010 |
| 4/7 | 16 | 111111110011011 |
| 4/8 | 16 | 111111110011100 |
| 4/9 | 16 | 111111110011101 |
| 4/A | 16 | 111111110011110 |
| 5/1 | 6 | 111011 |
| 5/2 | 10 | 1111111001 |
| 5/3 | 16 | 111111110011111 |
| 5/4 | 16 | 111111110100000 |
| 5/5 | 16 | 111111110100001 |
| 5/6 | 16 | 111111110100010 |
| 5/7 | 16 | 111111110100011 |
| 5/8 | 16 | 111111110100100 |
| 5/9 | 16 | 111111110100101 |
| 5/A | 16 | 111111110100110 |
| 6/1 | 7 | 1111001 |
| 6/2 | 11 | 11111110111 |
| 6/3 | 16 | 111111110100111 |
| 6/4 | 16 | 111111110101000 |
| 6/5 | 16 | 111111110101001 |
| 6/6 | 16 | 111111110101010 |
| 6/7 | 16 | 111111110101011 |
| 6/8 | 16 | 111111110101100 |
| 6/9 | 16 | 111111110101101 |
| 6/A | 16 | 111111110101110 |
| 7/1 | 7 | 1111010 |
| 7/2 | 11 | 11111111000 |
| 7/3 | 16 | 111111110101111 |
| 7/4 | 16 | 111111110110000 |
| 7/5 | 16 | 111111110110001 |
| 7/6 | 16 | 111111110110010 |
| 7/7 | 16 | 111111110110011 |
| 7/8 | 16 | 111111110110100 |
| 7/9 | 16 | 111111110110101 |
| 7/A | 16 | 111111110110110 |
| 8/1 | 8 | 11111001 |
| 8/2 | 16 | 111111110110111 |
| 8/3 | 16 | 111111110111000 |

Quadro A.4: Tabela de coeficientes AC de crominância(cont.).

| Run/Size | Code length | Code word |
|----------|-------------|-----------------|
| 8/4 | 16 | 111111110111001 |
| 8/5 | 16 | 111111110111010 |
| 8/6 | 16 | 111111110111011 |
| 8/7 | 16 | 111111110111100 |
| 8/8 | 16 | 111111110111101 |
| 8/9 | 16 | 111111110111110 |
| 8/A | 16 | 111111110111111 |
| 9/1 | 9 | 111110111 |
| 9/2 | 16 | 111111111000000 |
| 9/3 | 16 | 111111111000001 |
| 9/4 | 16 | 111111111000010 |
| 9/5 | 16 | 111111111000011 |
| 9/6 | 16 | 111111111000100 |
| 9/7 | 16 | 111111111000101 |
| 9/8 | 16 | 111111111000110 |
| 9/9 | 16 | 111111111000111 |
| 9/A | 16 | 111111111001000 |
| A/1 | 9 | 11111000 |
| A/2 | 16 | 111111111001001 |
| A/3 | 16 | 111111111001010 |
| A/4 | 16 | 111111111001011 |
| A/5 | 16 | 111111111001100 |
| A/6 | 16 | 111111111001101 |
| A/7 | 16 | 111111111001110 |
| A/8 | 16 | 111111111001111 |
| A/9 | 16 | 111111111010000 |
| A/A | 16 | 111111111010001 |
| B/1 | 9 | 11111001 |
| B/2 | 16 | 111111111010010 |
| B/3 | 16 | 111111111010011 |
| B/4 | 16 | 111111111010100 |
| B/5 | 16 | 111111111010101 |
| B/6 | 16 | 111111111010110 |
| B/7 | 16 | 111111111010111 |
| B/8 | 16 | 111111111011000 |
| B/9 | 16 | 111111111011001 |
| B/A | 16 | 111111111011010 |
| C/1 | 9 | 111111010 |
| C/2 | 16 | 111111111011011 |
| C/3 | 16 | 111111111011100 |
| C/4 | 16 | 111111111011101 |
| C/5 | 16 | 111111111011110 |

Quadro A.4: Tabela de coeficientes AC de crominância (cont.).

| Run/Size | Code length | Code word |
|-----------|-------------|------------------|
| C/6 | 16 | 1111111111011111 |
| C/7 | 16 | 1111111111100000 |
| C/8 | 16 | 1111111111100001 |
| C/9 | 16 | 1111111111100010 |
| C/A | 16 | 1111111111100011 |
| D/1 | 11 | 11111111001 |
| D/2 | 16 | 1111111111100100 |
| D/3 | 16 | 1111111111100101 |
| D/4 | 16 | 1111111111100110 |
| D/5 | 16 | 1111111111100111 |
| D/6 | 16 | 1111111111101000 |
| D/7 | 16 | 1111111111101001 |
| D/8 | 16 | 1111111111101010 |
| D/9 | 16 | 1111111111101011 |
| D/A | 16 | 1111111111101100 |
| E/1 | 14 | 1111111100000 |
| E/2 | 16 | 1111111111101101 |
| E/3 | 16 | 1111111111101110 |
| E/4 | 16 | 1111111111101111 |
| E/5 | 16 | 111111111110000 |
| E/6 | 16 | 111111111110001 |
| E/7 | 16 | 111111111110010 |
| E/8 | 16 | 111111111110011 |
| E/9 | 16 | 111111111110100 |
| E/A | 16 | 111111111110101 |
| F/0 (ZRL) | 10 | 111111010 |
| F/1 | 15 | 11111111000011 |
| F/2 | 16 | 111111111110110 |
| F/3 | 16 | 111111111110111 |
| F/4 | 16 | 11111111111000 |
| F/5 | 16 | 11111111111001 |
| F/6 | 16 | 11111111111010 |
| F/7 | 16 | 11111111111011 |
| F/8 | 16 | 11111111111100 |
| F/9 | 16 | 11111111111101 |
| F/A | 16 | 11111111111110 |

B. Formato dos Dados Comprimidos

Estruturalmente, os formatos dos dados consistem numa serie ordenada de parâmetros, marcadores e segmentos de marcadores. Devido a que essas partes consecutivas se apresentam com códigos alinhados ao byte, cada formato de dados comprimidos está formado por uma seqüência ordenada de bytes de 8 bits. Para cada byte define-se o bit mais significativo e o bit menos significativo [25].

- **Parâmetros**

Os parâmetros são números inteiros, com valores especificos relativos ao proceso de codificação, as características da imagen fonte e outras particularidades seleccionáveis pela aplicação. Aos parâmetros atribuem-se códigos de 4 bits, 1 byte o 2 bytes. Exceto para alguns grupos opcionais de parâmetros , esses codificam informação importante sem a qual o processo de decodificação não pode reconstruir adequadamente a imagem.

- **Marcadores**

Os marcadores utilizam-se para identificar as diversas partes estruturais dos formatos dos dados comprimidos. A maioria dos marcadores començam com segmentos marcadores, que possuem um grupo de parâmetros interrelacionado; alguns marcadores são independentes. Todos os marcadores tem-se atribuído códigos de dois bytes: um byte X'FF' seguido de um byte que é diferente de 0 ou de X'FF'. O segundo byte especifica-se no quadro B.1. Um asterisco (*) indica um marcador que é independente, ou seja, que não é o inicio de um segmento de marcador.

B.1: Atribuição de códigos de marcação.

| Code Assignment | Symbol | Description |
|---|---|--|
| Start Of Frame markers, non-differential, Huffman coding | | |
| X'FFC0' X'FFC1' X'FFC2' X'FFC3' | SOF ₀ SOF ₁ SOF ₂ SOF ₃ | Baseline DCT Extended sequential DCT Progressive DCT Lossless (sequential) |
| Start Of Frame markers, differential, Huffman coding | | |
| X'FFC5' X'FFC6' X'FFC7' | SOF ₅ SOF ₆ SOF ₇ | Differential sequential DCT Differential progressive DCT Differential lossless (sequential) |
| Start Of Frame markers, non-differential, arithmetic coding | | |
| X'FFC8' X'FFC9' X'FFCA' X'FFCB' | JPG SOF ₉ SOF ₁₀ SOF ₁₁ | Reserved for JPEG extensions Extended sequential DCT Progressive DCT Lossless (sequential) |
| Start Of Frame markers, differential, arithmetic coding | | |
| X'FFCD' X'FFCE' X'FFCF' | SOF ₁₃ SOF ₁₄ SOF ₁₅ | Differential sequential DCT Differential progressive DCT Differential lossless (sequential) |
| Huffman table specification | | |
| X'FFC4' | DHT | Define Huffman table(s) |
| Arithmetic coding conditioning specification | | |
| X'FFCC' | DAC | Define arithmetic coding conditioning(s) |
| Restart interval termination | | |
| X'FFD0' through X'FFD7' | RST _m * | Restart with modulo 8 count "m" |
| Other markers | | |
| X'FFD8' X'FFD9' X'FFDA' X'FFDB' X'FFDC' X'FFDD' X'FFDE' X'FFDF' X'FFE0' through X'FFEF' X'FFF0' through X'FFFD' X'FFFE' | SOI* EOI* SOS DQT DNL DRI DHP EXP APP _n JPG _n COM | Start of image End of image Start of scan Define quantization table(s) Define number of lines Define restart interval Define hierarchical progression Expand reference component(s) Reserved for application segments Reserved for JPEG extensions Comment |
| Reserved markers | | |
| X'FF01' X'FF02' through X'FFBF' | TEM* RES | For temporary private use in arithmetic coding Reserved |

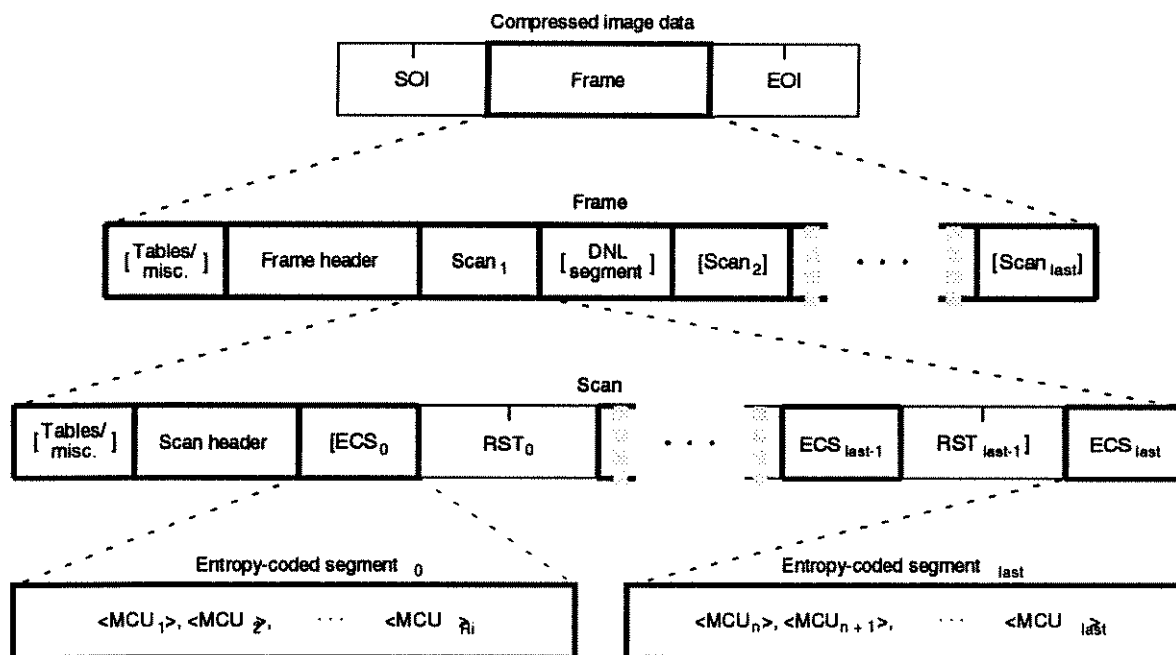
- **Segmento marcadores**

Um segmento marcador consiste num marcador seguido de uma seqüência de parâmetros com os que está relacionado. O primeiro parâmetro em um segmento marcador é o parâmetro de comprimento dos bytes.

- **Sintaxis**

O quadro B.2 mostra a sintaxis de alto nível. que especifica a ordem das partes consecutivas de alto nível do formato da informação codificada.

B.2: Sintaxis dos modos de operação sequencial baseado na DCT, progressivo baseado na DCT e sem perdas.



APÊNDICE

Frequências de Ocorrência dos Comprimentos das Palavras Código Computadas para a Determinação da Entropia.

- **smpte01.jpg:**

[comprimento:frequência:probabilidade]

2:1850:0.064234
3:8430:0.292698
4:4938:0.171452
5:3562:0.123676
6:3899:0.135377
7:2046:0.071039
8:1942:0.067428
9:57:0.001979
10:1162:0.040346
11:95:0.003298
12:117:0.004062
13:173:0.006007
14:3:0.000104
15:13:0.000451
16:72:0.002500
17:5:0.000174
18:82:0.002847
19:157:0.005451
20:103:0.003576
21:2:0.000069
22:93:0.003229
Entropia=2.966653

- **smpte02.jpg:**

[comprimento:frequência:probabilidade]

2:2044:0.073304
3:7847:0.281416
4:5041:0.180785
5:3246:0.116411
6:3788:0.135849
7:2103:0.075420
8:1865:0.066884
9:115:0.004124
10:904:0.032420
11:112:0.004017

12:147:0.005272
13:143:0.005128
14:20:0.000717
15:16:0.000574
16:67:0.002403
17:1:0.000036
18:88:0.003156
19:134:0.004806
20:111:0.003981
22:92:0.003299
Entropia=2.993091

- **smpte08.jpg:**

[comprimento:frequência:probabilidade]

2:3062:0.184658
3:4144:0.249910
4:2790:0.168255
5:1951:0.117658
6:1715:0.103425
7:1045:0.063020
8:691:0.041672
9:36:0.002171
10:429:0.025871
11:58:0.003498
12:138:0.008322
13:56:0.003377
14:2:0.000121
15:16:0.000965
16:39:0.002352
18:86:0.005186
19:128:0.007719
20:89:0.005367
21:16:0.000965
22:91:0.005488
Entropia=3.012826

- **smpte15.jpg:**

[comprimento:frequência:probabilidade]

3:7967:0.288158
4:4807:0.173864
5:3193:0.115488
6:3773:0.136466
7:2005:0.072519
8:2022:0.073134
9:125:0.004521
10:1006:0.036386
11:104:0.003762
12:138:0.004991
13:183:0.006619
14:11:0.000398

15:37:0.001338
16:87:0.003147
17:1:0.000036
18:87:0.003147
19:143:0.005172
20:129:0.004666
21:1:0.000036
22:92:0.003328
Entropia=3.008153

- **bar100.jpg:**

[comprimento:frequência:probabilidade]
2:7424:0.633880
3:576:0.049180
4:224:0.019126
5:320:0.027322
6:992:0.084699
7:352:0.030055
8:320:0.027322
9:384:0.032787
10:544:0.046448
11:288:0.024590
12:96:0.008197
13:192:0.016393
Entropia=2.130086

TRABALHOS APRESENTADOS

- CHUNG, F. e IANO, Y. “Aspectos Relevantes na Codificação de Imagens segundo o Padrão JPEG.”, <to appear>, VI-RPIC, Argentina, 1995.
- CHUNG, F. e IANO, Y. “Investigações Preliminares do Padrão JPEG para Codificação de Imagens com Wavelets.”, <to appear>, ELECTRO’95, Chile, 1995.