

Notas de Aula: Multimídia

Compressão JPEG

Autor: Rudinei Goularte
{rudinei}@icmc.usp.br

Instituto de Ciências Matemáticas e de Computação (ICMC)
Universidade de São Paulo, São Carlos - SP

Resumo

Este texto apresenta algumas informações condensadas sobre o processo de compressão de imagens no padrão JPEG, no contexto das disciplinas do ICMC voltadas para a área de Multimídia. O objetivo é facilitar o entendimento, servir como suporte inicial aos estudos desse tema assim como de suporte para o desenvolvimento de um compressor JPEG.

1 Compressão JPEG

Entre 1985 e 1990, foi organizado um comitê conjunto da ISO (*International Organization for Standardization*) e do CCITT (*International Telegraph and Telephone Consultative Committee*), hoje ITU-T, parte da ITU (*International Telecommunications Union*), com a finalidade de desenvolver um padrão de compressão de "imagens do mundo real". Este comitê recebeu a denominação de JPEG (*Joint Photographic Experts Group*). O que é conhecido hoje como padrão JPEG é um documento (também chamado de norma), denominado oficialmente como Recomendação JTC1 / 10918-2, o qual normatiza vários processos de compressão para imagens estáticas, incluindo um método de compactação sem perdas. O método sequencial "com perdas" é o mais utilizado, e é baseado na Transformada Discreta de Cossenos (DCT), seguido

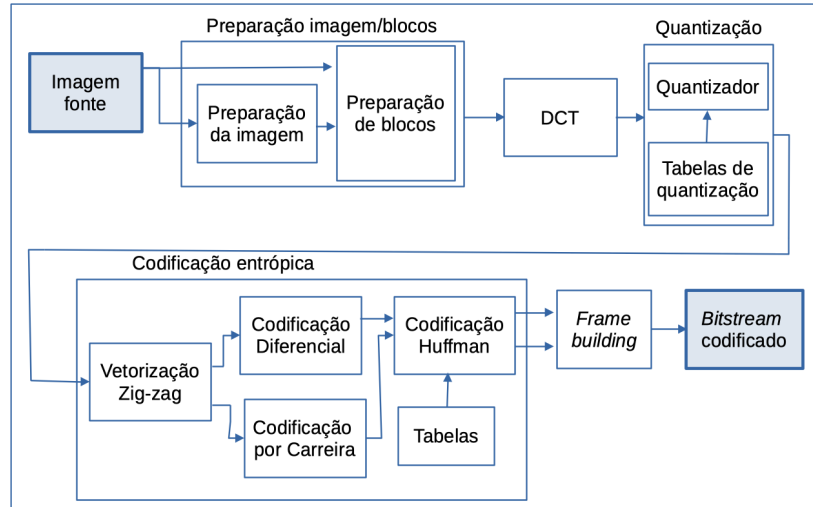


Figura 1: Esquema em diagrama de blocos do processo de codificação sequencial do padrão JPEG. Fonte: autor.

de quantização variável e codificação entrópica. A Figura 1 ilustra o processo do JPEG sequencial.

A **DCT** decompõe um bloco da imagem original de 8x8 pixels na forma de uma soma ponderada de funções-base, ou primitivas, que são ortogonais entre si. Nesta etapa, a informação espacial contida na imagem é analisada e reagrupada em coeficientes que são relacionados ao conteúdo espectral da imagem (chamado de frequência espacial). Ou seja, a DCT transforma informação no domínio espacial (dos pixels) em informação no domínio das frequências (espaciais). O bloco transformado possui valores organizados de acordo com a intensidade das frequências - das mais baixas para as mais altas. O primeiro valor do bloco (posição 0,0 da matriz 8x8) é chamada de coeficiente DC. Os outros 63 valores são chamados de coeficientes AC. Essa organização é extremamente útil, pois o olho humano não capta facilmente diferenças nas altas frequências espaciais. Desse modo, em etapas subsequentes, parte das informações em alta frequência pode ser eliminada.

As perdas de informação e dados ocorrem efetivamente na etapa de quantização, onde a taxa de compressão pode ser aumentada, em detrimento da

qualidade, ajustando-se um fator de escala no quantizador. Esse fator de escala é aplicado a uma tabela de quantização de dimensão 8x8, onde cada valor divide o valor correspondente no bloco proveniente da DCT.

A etapa de codificação entrópica é dividida em 4 sub-etapas. A primeira é a vetorização em zig-zag, que consiste em transformar cada bloco (matriz) de 8x8 posições em um vetor (linear) de 64 posições. O modo de percorrer o bloco segue uma lógica em zig-zag, tentando cobrir os valores na ordem da menor frequência para a maior. Assim, espera-se que no final do vetor estejam concentrados os valores zero (0). A codificação R. L. E. (*Run Length Encoding*) e a codificação Huffman otimizam a quantidade de bits usados para transmitir a informação quantizada. A R.L.E. é aplicada ao conjunto de vetores (cada um corresponde a um bloco) de toda a imagem, de modo a considerar os 63 últimos valores (exclui-se o primeiro valor) de cada vetor. Ao conjunto formado pelo primeiro valor de cada vetor é aplicada a Codificação Diferencial. Os resultados da R.L.E. e da Codificação Diferencial passam por Codificação Huffman, a qual utiliza tabelas de códigos pré-definidas para gerar códigos de tamanho variável para cada valor pré-codificado por R.L.E. e por Diferenças.

Deve-se observar que o padrão JPEG foi desenvolvido para atender à compressão de imagens de tom contínuo, isto é imagens “do mundo real”, com tonalidades contínuas, como as obtidas por meio de câmeras fotográfica e celulares. Por esse motivo o padrão JPEG não é adequado para imagens vetoriais, geométricas ou de alto contraste, como textos impressos, desenhos técnicos, desenhos geométricos, etc.

1.1 Transformada DCT

Para imagens coloridas, no espaço de cor RGB, a etapa de Preparação de Imagem realiza a conversão para o espaço de cor YCbCr (derivado do modelo YUV). As componentes R, G e B são mapeadas para as componentes Y, Cb e Cr de acordo com equações padronizadas por normas técnicas. Uma dessas normas é a ITU-R BT-601:

$$Y = 0.299R + 0.587G + 0.114B \quad (1)$$

$$Cb = 0.564(B - Y) \quad (2)$$

$$Cr = 0.713(R - Y) \quad (3)$$

$$R = Y + 1.402Cr \quad (4)$$

$$G = Y - 0.344Cb - 0.714Cr \quad (5)$$

$$B = Y + 1.772Cb \quad (6)$$

As componentes Cb e Cr podem ser sub-amostradas de modo a reduzir a resolução espacial pela metade, nos sentidos vertical e horizontal (subamostragem (*Downsampling*) 4:2:0). A imagem é então dividida em blocos de 8x8 pixels, seja no plano da luminância como nos da crominância, sendo que estes blocos são processados independentemente uns dos outros. O dimensionamento dos blocos (8x8) leva em conta a resposta em frequência espacial da visão humana: supondo que a taxa de amostragem espacial foi adequadamente escolhida, a distância entre 2 pixels deve estar perto do limite de resolução visual (1 a 2 minutos de grau). Nestas condições, a dimensão do bloco corresponde a detalhes de aproximadamente 8 a 16 minutos de grau, para os quais a visão humana atinge sensibilidade máxima.

A transformada DCT de um bloco de imagem $f(x,y)$ é definida por:

$$F[i, j] = \frac{1}{4}C(i)C(j) \sum_{x=0}^7 \sum_{y=0}^7 P[x, y] \cos \frac{(2x+1)i\pi}{16} \cos \frac{(2y+1)j\pi}{16} \quad (7)$$

onde

$$C_{i,j} = \begin{cases} \frac{1}{\sqrt{2}}, & \text{para } i \text{ e } j = 0; \\ 1, & \text{para todos os outros valores de } i \text{ e } j. \end{cases} \quad (8)$$

Deve-se observar que, por se tratar de um bloco de 8x8 pixels, x , y , i e j variam de 0 a 7. Outra observação diz respeito à constante $\frac{1}{4}$. Ela deriva do caso geral $(\frac{2}{x})^{\frac{1}{2}} \cdot (\frac{2}{y})^{\frac{1}{2}}$ em que x e y são as dimensões de uma imagem ou bloco. Quando restringimos x e y para 8, o resultado é $\frac{1}{4}$.

A transformada DCT possui uma inversa, que leva os valores no domínio das frequências espaciais de volta para o domínio dos pixels. Matematicamente sem perdas. Computacionalmente podem ocorrer perdas devido a erros de arredondamento. A função inversa da DCT é definida por:

.354	.354	.354	.354	.354	.354	.354	.354
.490	.416	.278	.098	-.098	-.278	-.416	-.490
.462	.191	-.191	-.462	-.462	-.191	.191	.462
.416	-.098	-.490	-.278	.278	.490	.098	-.416
.354	-.354	-.354	.354	.354	-.354	-.354	.354
.278	-.490	.098	.416	-.416	-.098	.490	-.278
.191	-.462	.462	-.191	-.191	.462	-.462	.191
.098	-.278	.416	-.490	.490	-.416	.278	-.098

Tabela 1: Matriz de transformação C para o produto matricial da DCT separável. Fonte: Internet.

$$P[x, y] = \frac{1}{4} \sum_{x=0}^7 \sum_{y=0}^7 C(i)C(j)F[i, j] \cos \frac{(2x+1)i\pi}{16} \cos \frac{(2y+1)j\pi}{16} \quad (9)$$

A DCT é separável, ou seja, pode ser efetuada sequencialmente, primeiro no sentido horizontal e depois no vertical. Um modo de implementar a DCT utilizando esta propriedade é por meio de um produto matricial: $DCT = C \times B \times C^T$ onde B é um bloco de 8 x 8 pixels da imagem original e C é uma matriz de transformação, definida por:

$$C(i, j) = \begin{cases} \frac{1}{\sqrt{8}}, & \text{se } i = 0; \\ \frac{1}{2} \cos\left(\frac{(2j+1)i\pi}{16}\right), & \text{se } i > 0. \end{cases} \quad (10)$$

Desse modo, pode-se pré-calculer a matriz C para um bloco 8x8 (Tabela 1):

1.2 Quantização

Os coeficientes resultantes da DCT são quantizados usando tabelas pré-definidas $Q(u, v)$ e arredondados para o inteiro mais próximo:

$$F_Q(u, v) = \text{round} \left\{ \frac{F(u, v)}{k \cdot Q(u, v)} \right\} \quad (11)$$

onde k = fator de compressão.

O JPEG fornece 2 tabelas de quantização (Figura 2) padronizadas, uma para a componente de Luminância (Y) e outra para as componentes de Crominância (Cb e Cr). Os coeficientes apresentados foram obtidos experimentalmente, e levam em conta a resposta visual em frequência espacial para detalhes de luminância e crominância. As tabelas de quantização podem ser definidas pelo usuário durante a compressão da imagem, a depender da implementação do compressor, e, nesse caso, devem ser anexadas ao arquivo comprimido.

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
79	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Luminância

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Crominância

Figura 2: Tabelas de quantização sugeridas pelo padrão JPEG. Fonte: Wikipedia (<https://en.wikipedia.org/wiki/JPEG>)

Após a quantização, a tendência para imagens típicas é de que a maior parte dos coeficientes seja zero, restando apenas alguns coeficientes de baixa frequência espacial (concentrados no canto superior esquerdo). Os coeficientes quantizados são reordenados em um vetor seguindo um padrão em zigue-zague (Figura 3), que tende a agrupar os coeficientes de maior importância no início e os zeros no fim do vetor (Figura 4).

1.3 Codificação Entrópica

Cada bloco da imagem, agora na forma de um vetor de coeficientes DCT quantizados, passa pela Codificação Entrópica (também chamada de Codificação por Entropia). Os coeficientes DC, $F(0, 0)$ de cada bloco quantizado, ou

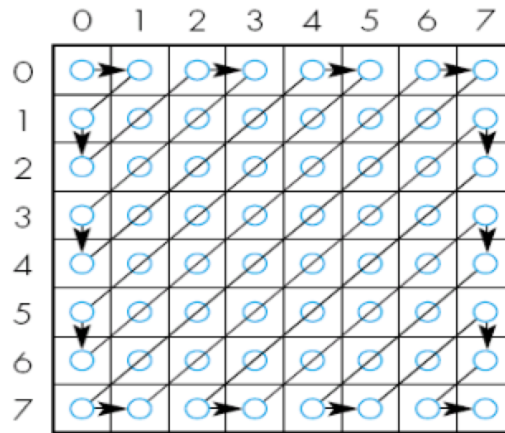


Figura 3: Zig-Zag scan dos coeficientes quantizados. Fonte: autor.

0	1	2	3	4	5	6	7	8	9	10	11	12				63
512	16	27	3	3	3	2	2	2	2	0	0	0		0

Figura 4: Vetor resultante do zig-zag scan. Valores fictícios. A posição 0 contém o coeficiente DC, de mais baixa frequência. Os demais 63 valores são coeficientes AC, ordenados pela frequência espacial da mais baixa para a mais alta. Os zeros tendem a se concentrar no fim do vetor. Fonte: autor.

agora na posição 0 de cada vetor, são processados separadamente. Os demais coeficientes (AC) passam por codificação RLE (*Run-Length Encoding*), onde são criados códigos consistindo do par ordenado:

(*quantidade de zeros / valor do coeficiente*)

Quando na sequência reordenada de símbolos não há mais coeficientes não nulos dentro do bloco, a sequência é terminada por um símbolo especial (EOB - *End of Block*). É usada finalmente uma codificação Huffman modificada, para traduzir os símbolos gerados pela RLE em padrões binários de comprimento variável. Nesta codificação, os valores dos coeficientes são representados por uma categoria e uma mantissa:

- a categoria especifica a ordem de grandeza do coeficiente e define a quantidade de bits necessária para representar a mantissa;
- a mantissa fornece o valor do coeficiente, dentro da faixa correspondente

à categoria. Os símbolos assumem então a forma:
 $\{quantidade\ de\ zeros/categoria/mantissa\}$.

O código de Huffman utiliza uma tabela para codificar o par ordenado $\{quantidade\ de\ zeros + categoria\}$ como um código de comprimento variável, composto de um prefixo e de uma mantissa, cujo comprimento está implícito no prefixo. A taxa de compressão efetiva do JPEG não é conhecida *a priori*, pois depende dos parâmetros estatísticos específicos da imagem. Atuando no parâmetro k (fator de compressão) conseguem-se taxas de compressão de 20:1 ou mais, mantendo degradação aceitável.

2 Exemplo de Codificação JPEG

Dado o bloco de 8x8 pixels extraído de uma imagem real ilustrado na Figura 5:

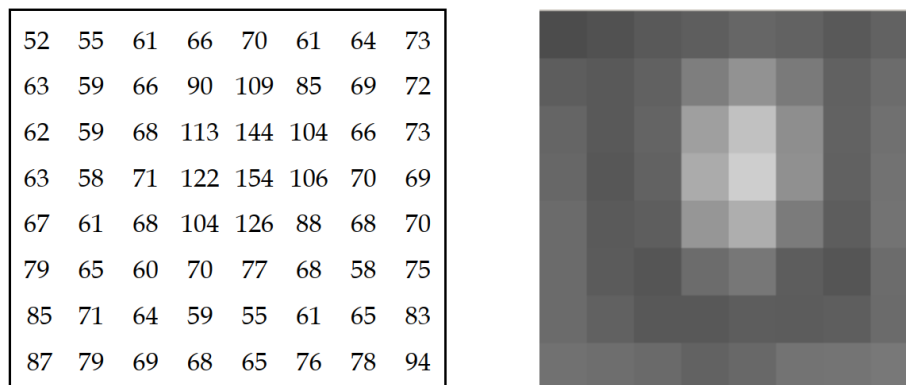


Figura 5: Blocos de 8x8 pixels extraído de uma imagem real. À esquerda o bloco com os valores numéricos de cada pixel. À direita a visualização do bloco e dos pixels. Fonte: Wikipedia (<https://en.wikipedia.org/wiki/JPEG>)

Após aplicar a DCT a este bloco e em seguida a quantização utilizando a tabela ilustrada na Figura 2, tem-se como resultado os valores ilustrados na Figura 6.

Após a reordenação em zigue-zague, o tem-se o seguinte resultado:

[-26 -3 1 -3 -2 -6 2 -4 1 -4 1 1 5 0 2 0 0 -1 2 0 0 0 0 0 -1 -1 EOB]

-415	-29	-62	25	55	-20	-1	3
7	-21	-62	9	11	-7	-6	6
-46	8	77	-25	-30	10	7	-5
-50	13	35	-15	-9	6	0	3
11	-8	-13	-2	-1	1	-4	1
-10	1	3	-3	-1	0	2	-1
-4	-1	2	-1	2	-3	1	-2
-1	-1	-1	-2	-1	-1	0	-1

Coeficientes da DCT

-26	-3	-6	2	2	0	0	0
1	-2	-4	0	0	0	0	0
-3	1	5	-1	-1	0	0	0
-4	1	2	-1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Coeficientes DCT quantizados

Figura 6: Coeficientes DCT e coeficientes quantizados. Fonte: Adaptado de [1].

O primeiro coeficiente (-26), coeficiente DC, representa a intensidade média de luminância do bloco. Este coeficiente é codificado por uma tabela própria, sendo na realidade processada a diferença entre este coeficiente e o coeficiente DC do bloco de imagem imediatamente anterior (Codificação Diferencial). Este coeficiente deve ser codificado com a melhor precisão possível, uma vez que qualquer descontinuidade entre blocos vizinhos torna-se bastante visível. Supondo que este bloco é o primeiro de uma imagem, então, pela Tabela 2, o valor - 26 pertence à categoria 5, sendo codificado (Tabela 3) por um padrão 110 00101. Cada categoria indica implicitamente a quantidade de bits do valor. Então, temos o prefixo 110 da categoria 5, seguido de 5 bits que representam o valor -26: 00101 é -26 em complemento de 1.

Os demais coeficientes (coeficientes "AC", representando frequências espaciais não nulas) são agrupados em pares através de codificação RLE, obtendo-se:

[0/-3, 0/1, 0/-3, 0/-2, 0/-6, 0/2, 0/-4, 0/1, 0/-4, 0/1, 0/1, 0/5, 1/2, 2/-1, 0/2, 5/-1, 0/-1, EOB]

Os valores dos coeficientes são classificados em categorias de acordo com sua ordem de grandeza, conforme a Tabela 2. Assim sendo, temos:

[0/2/-3, 0/1/1, 0/2/-3, 0/2/-2, 0/3/-6, 0/2/2, 0/3/-4, 0/1/1, 0/3/-4, 0/1/1, 0/1/1, 0/3/5, 1/2/2, 2/1/-1, 0/2/2, 5/1/-1, 0/1/-1, EOB]

Consultando o "Codebook"JPEG (Tabela 4), indexado pelo par {zeros / ca-

tegoria}, os símbolos acima, incluindo o coeficiente DC já codificado, são convertidos para o seguinte **conjunto de bits comprimidos** representando o bloco:

11000101 0100 001 0100 0101 100001 0110 100011 001 100011 001 001 100101
11100110 110110 0110 11110100 000 1010

Note-se que os prefixos (em destaque em vermelho) são concatenados ao valor da mantissa (código binários em complemento de 1 do valor do coeficiente). No final o bloco original (512 bits (8x8 pixels x 8 bits/pixel)) foi representado por uma sequência de 93 bits, o que significa uma taxa de compressão de 5.5 : 1, ou 1.45 bits/pixel.

O bloco analisado é um exemplo de imagem "difícil" para o JPEG, pois contém um pico de luminância restrito ao centro do bloco. Em situações normais, a maior parte dos blocos apresentará apenas 2 ou 3 coeficientes quantizados, proporcionando taxas de compressão muito maiores.

A Figura 7 mostra o bloco reconstruído a partir dos coeficientes quantizados, bem como o erro de reconstrução (diferença para o bloco original). Para este exemplo, o erro R.M.S. é da ordem de 5.9 níveis de intensidade.

Apesar dos erros de reconstrução, o processo preservou os elementos de maior importância dentro do bloco. A diferença pode ser imperceptível, considerando que a imagem seja visualizada em condições normais, ou seja, onde o tamanho aparente de cada pixel esteja no limite da acuidade visual.

Quando o fator de compressão é muito elevado, o processo JPEG apresenta um efeito de descontinuidade nos contornos dos blocos, denominado de "blocagem", causado pelo fato de que o tratamento dado a cada bloco é independente dos demais. A Figura 8 ilustra o efeito de blocagem em uma imagem reconstruída comparada à original.

O conjunto de bits comprimidos (o bloco comprimido) descrito acima pode ser revertido para os valores originais do Bloco. Como cada prefixo é um código Huffman, pode-se obter via Tabela 4 a correspondente quantidade de zeros que antecede o valor do coeficiente e a quantidade de bits a ser lida em seguida, a qual representa o próprio valor do coeficiente. A partir daí, as etapas lineares do JPEG são seguidas ao contrário, aplicando-se as respectivas funções inversas de cada etapa (Decodificação R.L.E. e Decod. Diferencial, zig-zag scan reverso, Dequantização, DCT Inversa, ...).

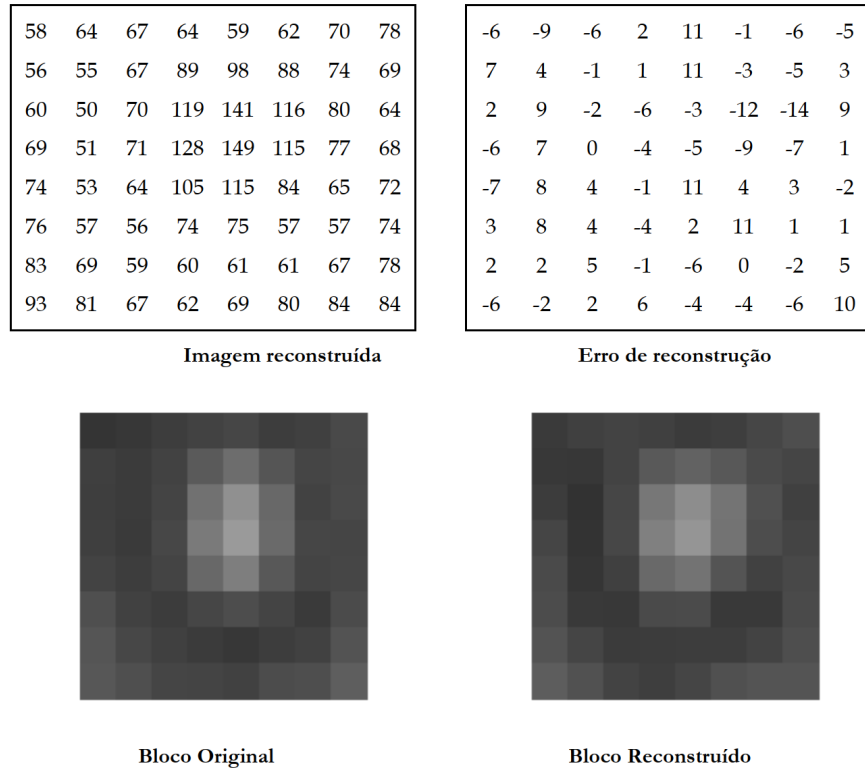


Figura 7: Blocos original e reconstruído. As diferenças existem, mas são quase imperceptíveis em condições normais de visualização. Fonte: Adaptado de [1].

3 Tabelas JPEG para Codificação de Imagens

A representação dos coeficientes quantizados no JPEG, por codificação entrópica, é composta de um prefixo de comprimento variável, seguido por um campo de codificação do valor do coeficiente (também de comprimento variável). Os prefixos definem categorias de comprimento para os símbolos/valores codificadas, e correspondem à quantidade de bits necessária para representação do valor do coeficiente quantizado. Por exemplo, a categoria 5 reserva 5 bits para codificar os 32 valores compreendidos no intervalo $[-31$ a -16 e $+16$ a $+31]$. A Tabela 2 mostra as categorias correspondentes a cada faixa de valores representáveis dos coeficientes (DC ou AC). Podem ser representados coeficientes DC de -2047 a $+2047$, e coeficientes AC de -1023



Figura 8: Efeito de blocagem. Na imagem à direita pode-se observar a presença de "quadriculados", não presentes na imagem original (à esquerda). Fonte: Adaptado de [1].

a -1 e de +1 a +1023 (o zero não é representado).

Faixa de valores	Categoria DC	Categoria AC
0	0	-
-1,1	1	1
-3, -2, 2, 3	2	2
-7,...,-4, 4, ..., 7	3	3
-15, ..., -8, 8, ..., 15	4	4
-31, ..., -16, 16, ..., 31	5	5
-63, ..., -32, 32, ..., 63	6	6
-127, ..., -64, 64, ..., 127	7	7
-255, ..., -128, 128, ..., 255	8	8
-511, ..., -256, 256, ..., 511	9	9
-1023, ..., -512, 512, ..., 1023	A	A
-2047, ..., -1024, 1024, ..., 2047	B	-

Tabela 2: Categorias para codificação de coeficientes DC e AC no padrão JPEG. Fonte: [1].

A Tabela 3 fornece os prefixos e o comprimento total dos símbolos codificados para as categorias correspondentes aos coeficientes DC (correspondentes a $F(0,0)$ após quantização). Por exemplo: um coeficiente DC de valor +200 é

Categoria	Prefixo	Comprimento	Qtd. bits da mantissa
0	010	3	0
1	011	4	1
2	100	5	2
3	00	5	3
4	101	7	4
5	110	8	5
6	1110	10	6
7	11110	12	7
8	111110	14	8
9	1111110	16	9
A	11111110	18	10

Tabela 3: Prefixos para Coeficientes DC. Fonte: [1]

representável pela categoria 8 (que cobre a faixa de valores de -255 a -128 e de +128 a +255). Na Tabela 3, vemos que o prefixo correspondente é 111110 (possui 6 bits) e o comprimento total é 14 bits; restam portanto 8 bits para codificar o valor 200 dentro da faixa coberta por aquela categoria. Dentro desta faixa os valores são codificados em *Complemento de 1* da seguinte maneira:

Valor	Código
-255	00000000
-254	00000001
-253	00000010
...	
-128	01111111
+128	10000000
+129	10000001
...	
+255	11111111

Desta forma, o valor 200 será codificado como 11001000; concatenando-o com o prefixo, temos então que $+200 = 111110\ 11001000$.

Já para os coeficientes AC, a Tabela 4 traz, para cada combinação de {quantidade de zeros / categoria}, o correspondente prefixo, e o comprimento total do símbolo. Por exemplo, para uma sequência de 2 coeficientes nulos seguidos de um coeficiente igual a -17 (ou seja, $\{ 2 / 5 / -17 \}$) temos, pela Tabela 2, que a categoria 5 é adequada para representar o valor -17, através de

uma mantissa de 5 bits. Esta mantissa será representada por uma sequência 01110. Na Tabela 4, o prefixo correspondente à combinação (2 / 5), ou seja, dois zeros e um valor da categoria 5, é 1111111110001010. Acrescentando a mantissa do coeficiente, chegamos a 1111111110001010 01110.

Zeros/categoria	Prefixo	Compr. total	Zeros/categoria	Prefixo	Compr. total
0/0	1010 (=EOB)	4	8/1	11111010	9
0/1	00	3	8/2	11111111000000	17
0/2	01	4	8/3	111111110110111	19
0/3	100	6	8/4	111111110111000	20
0/4	1011	8	8/5	111111110111001	21
0/5	11010	10	8/6	111111110111010	22
0/6	111000	12	8/7	111111110111011	23
0/7	1111000	14	8/8	111111110111100	24
0/8	1111110110	18	8/9	111111110111101	25
0/9	111111110000010	25	8/A	111111110111110	26
0/A	111111110000011	26	9/1	1111111000	10
1/1	1100	5	9/2	111111110111111	18
1/2	111001	8	9/3	111111111000000	19
1/3	1111001	10	9/4	111111111000001	20
1/4	111110110	13	9/5	111111111000010	21
1/5	11111110110	16	9/6	111111111000011	22
1/6	111111110000100	22	9/7	1111111110000100	23
1/7	111111110000101	23	9/8	1111111110000101	24
1/8	111111110000110	24	9/9	1111111110000110	25
1/9	111111110000111	25	9/A	1111111110000111	26
1/A	111111110001000	26	A/1	1111111001	10
2/1	11011	6	A/2	111111111001000	18
2/2	11111000	10	A/3	1111111111001001	19
2/3	1111110111	13	A/4	1111111111001010	20
2/4	111111110001001	20	A/5	1111111111001011	21
2/5	1111111110001010	21	A/6	1111111111001100	22
2/6	1111111110001011	22	A/7	1111111111001101	23
2/7	1111111110001100	23	A/8	1111111111001110	24
2/8	1111111110001101	24	A/9	1111111111001111	25
2/9	1111111110001110	25	A/A	111111111010000	26
2/A	1111111110001111	26	B/1	111111010	10
3/1	111010	7	B/2	1111111111010001	18
3/2	111110111	11	B/3	1111111111010010	19
3/3	11111110111	14	B/4	1111111111010011	20
3/4	111111110010000	20	B/5	1111111111010100	21
3/5	111111110010001	21	B/6	1111111111010101	22
3/6	111111110010010	22	B/7	1111111111010110	23
3/7	111111110010011	23	B/8	1111111111010111	24
3/8	111111110010100	24	B/9	1111111111010100	25
3/9	111111110010101	25	B/A	1111111111010101	26
3/A	111111110010110	26	C/1	1111111010	11
4/1	111011	7	C/2	111111111101010	17
4/2	111111000	12	C/3	1111111111010111	19
4/3	111111110010111	19	C/4	1111111111011100	20
4/4	1111111110011000	20	C/5	1111111111011101	21
4/5	1111111110011001	21	C/6	1111111111011110	22
4/6	1111111110011010	22	C/7	1111111111011111	23
4/7	1111111110011011	23	C/8	1111111111000000	24
4/8	1111111110011100	24	C/9	1111111111000001	25
4/9	1111111110011101	25	C/A	1111111111000010	26
4/A	1111111110011110	26	D/1	11111111010	12
5/1	1111010	8	D/2	1111111111100011	18
5/2	1111111001	12	D/3	1111111111100100	19
5/3	111111111001111	19	D/4	1111111111100101	20
5/4	1111111110100000	20	D/5	1111111111100110	21
5/5	1111111110100001	21	D/6	1111111111100111	22
5/6	1111111110100010	22	D/7	1111111111101000	23
5/7	1111111110100011	23	D/8	1111111111101001	24
5/8	1111111110100100	24	D/9	1111111111101010	25
5/9	1111111110100101	25	D/A	111111111101011	26
5/A	1111111110100110	26	E/1	111111110110	13
6/1	1111011	8	E/2	1111111111101100	18
6/2	111111111000	13	E/3	1111111111101101	19
6/3	1111111110100111	19	E/4	1111111111101110	20
6/4	1111111110101000	20	E/5	1111111111101111	21
6/5	1111111110101001	21	E/6	1111111111100000	22
6/6	1111111110101010	22	E/7	1111111111100001	23
6/7	1111111110101011	23	E/8	111111111110010	24
6/8	1111111110101100	24	E/9	111111111110011	25
6/9	1111111110101101	25	E/A	111111111110100	26
6/A	1111111110101110	26	F/0	111111110111 (*)	12
7/1	11111001	9	F/1	111111111110101	17
7/2	1111111001	13	F/2	111111111110110	18
7/3	1111111110101111	19	F/3	111111111110111	19
7/4	1111111110110000	20	F/4	111111111111000	20
7/5	1111111110110001	21	F/5	1111111111110001	21
7/6	1111111110110010	22	F/6	111111111111010	22
7/7	1111111110110011	23	F/7	111111111111011	23
7/8	1111111110110100	24	F/8	111111111111100	24
7/9	1111111110110101	25	F/9	111111111111101	25
7/A	1111111110110110	26	F/A	111111111111110	26

Tabela 4: Prefixos para Coeficientes AC. (*) Extensão de zeros. Fonte: Adaptado de [1] e [3].

Referências

- [1] R.C. Gonzalez and R.E. Woods, R.E., *Digital Image Processing*, isbn=9780133356724, Pearson, 2018.
- [2] D. Salomon, *Data Compression: The Complete Reference*, isbn=9780387218328, Springer New York, 2006.
- [3] W.B. Pennebaker, and J.L. Mitchell, *JPEG: Still Image Data Compression Standard*, isbn=9780442012724, Springer US, 1992.