

QUANTIZAÇÃO DA INFORMAÇÃO PARALELA EM CODIFICADORES DE IMAGEM BTC

Alberto Gaspar Guimarães
Instituto Militar de Engenharia
Praça General Tibúrcio, 80
22290-270, Rio de Janeiro-RJ
gaspar@aquarius.ime.eb.br

Abraham Alcaim
CETUC/PUC-Rio
Rua Marquês de São Vicente, 225
22453-900, Rio de Janeiro-RJ
alcaim@cetuc.puc-rio.br

RESUMO

Na codificação BTC ("Block Truncation Coding") a imagem é dividida em blocos, que são representados por um plano de bits e por uma informação paralela. O objetivo deste trabalho é comparar as diferentes técnicas BTC quando submetidas a esquemas de quantização distintos dos parâmetros da informação paralela. Nesta análise considerou-se tanto a quantização escalar, como a quantização vetorial aplicada separadamente a cada parâmetro. Resultados de simulação são apresentados, permitindo apontar as melhores alternativas em cada caso.

1. INTRODUÇÃO

A codificação em bloco por truncamento (BTC) tem sido aplicada com sucesso para compressão de imagens [1]-[7] devido à sua simplicidade, rapidez e facilidade de implementação. A técnica BTC usa um quantizador de 1 bit localmente adaptativo. Os níveis de quantização são ajustados de modo que certos momentos estatísticos de pequenos blocos de imagens de entrada sejam preservados na saída quantizada. No algoritmo BTC original [1] os níveis de quantização são escolhidos de forma a preservar a média \bar{X} e o desvio padrão σ do bloco. Outros esquemas de codificação foram propostos a partir deste, empregando diferentes conjuntos de parâmetros estatísticos como informação paralela [2]-[6]. Um esquema freqüentemente utilizado é o que preserva o primeiro momento absoluto central do bloco, cujo desempenho é ótimo no sentido do erro médio quadrático para imagens codificadas em BTC.

Embora a técnica BTC tenha muitos atrativos, observa-se que ela apresenta uma taxa de bits relativamente alta quando os parâmetros são quantizados de forma escalar. No algoritmo BTC original, por exemplo, codifica-se a imagem utilizando 6 bits para a média \bar{X} e 4 bits para σ , o que resulta em uma taxa de bits de 1,63 bit/pixel. Algoritmos propostos mais recentemente objetivam reduzir esta taxa mantendo o mesmo nível de desempenho, mas não conseguem taxas inferiores a 1,50 bit/pixel.

A alternativa normalmente empregada para melhorar a taxa de compressão é a utilização de esquemas de quantização

vetorial, principalmente aplicados à informação paralela [5],[7]. A complexidade computacional aumenta mas é possível atingir uma redução significativa da taxa de bits.

Os esquemas BTC implementados empregam diferentes parâmetros estatísticos como informação paralela. Além disso, diferentes técnicas de quantização podem ser aplicadas para sua codificação. Mostra-se oportuno, portanto, a realização de uma comparação de desempenho entre estas alternativas. Este é o objetivo do presente trabalho, que aborda, especificamente, três esquemas BTC utilizando dois tipos de quantização da informação paralela.

O artigo está organizado em cinco seções. Na Seção 2 são descritos sucintamente alguns métodos de codificação BTC. Na Seção 3 são apresentadas as alternativas para quantização da informação paralela em blocos BTC. Na Seção 4 os resultados obtidos em simulação são apresentados e analisados. Por fim, na Seção 5, são apresentadas as conclusões do trabalho.

2. ALGORITMOS BTC

Em BTC a imagem é dividida em blocos contendo $M=m_x m_y$ pixels, onde m é normalmente igual a quatro. Cada pixel X_i ($i=1, \dots, M$) em um bloco é quantizado para o valor Y_0 se ele for menor que um valor limiar X_{th} , e para Y_1 caso contrário. Aos níveis de quantização Y_0 e Y_1 são associados os bits 0 e 1, respectivamente. Portanto, cada bloco é representado por um plano de bits de tamanho $M = m_x m_y$ (aqui será utilizado $m=4$), juntamente com uma informação paralela relacionada aos parâmetros estatísticos do bloco.

Conforme mencionado anteriormente, são empregados diferentes parâmetros para compor a informação paralela no algoritmo BTC. A média \bar{X} definida por

$$\bar{X} = \frac{1}{M} \sum_{i=1}^M X_i \quad (1)$$

é em geral utilizada como um dos parâmetros de informação paralela e normalmente representa o valor limiar na definição do plano de bits. Para a escolha do segundo parâmetro há diferentes propostas. Os conjuntos de parâmetros analisados

neste trabalho consistem de $\{\bar{X}, \sigma\}$, $\{\bar{X}, \alpha\}$ e $\{\bar{X}, n\}$, ou seja, a média em conjunto com o desvio padrão, momento absoluto central e parâmetro n, respectivamente. As equações de definição destes parâmetros são apresentadas a seguir, juntamente com os níveis de quantização BTC a eles associados.

I. Método $\{\bar{X}, \sigma\}$

O desvio padrão é definido por

$$\sigma = \sqrt{\overline{X^2} - (\bar{X})^2} \quad (2)$$

onde

$$\overline{X^2} = \frac{1}{M} \sum_{i=1}^M X_i^2 \quad (3)$$

Os níveis de quantização Y_0 e Y_1 que preservam \bar{X} e σ são dados por

$$Y_0 = \bar{X} - \sigma \sqrt{\frac{M-\beta}{\beta}} \quad (4)$$

$$Y_1 = \bar{X} + \sigma \sqrt{\frac{\beta}{M-\beta}} \quad (5)$$

onde β é o número de pixels abaixo do limiar.

II. Método $\{\bar{X}, \alpha\}$

O momento absoluto central é definido por

$$\alpha = \frac{1}{M} \sum_{i=1}^M |X_i - \bar{X}| \quad (6)$$

Pode-se mostrar que se $X_{th} = \bar{X}$, então preservar o momento absoluto central é equivalente a preservar a média inferior (\bar{X}_L), definida como a média do conjunto formado pelos pixels com valores abaixo do limiar. Este quantizador é ótimo no sentido do erro médio quadrático.

Os valores de Y_0 e Y_1 para este quantizador são dados por

$$Y_0 = \bar{X} - \frac{M}{2\beta} \alpha \quad (7)$$

$$Y_1 = \bar{X} + \frac{M}{2(M-\beta)} \alpha \quad (8)$$

III. Método $\{\bar{X}, n\}$

Uma proposta recente, apresentada em [6], objetiva preservar a média \bar{X} , a média inferior \bar{X}_L e a média superior \bar{X}_H , estas últimas definidas como a média do conjunto de pixels com valores abaixo e acima do limiar, respectivamente. Os valores de Y_0 e Y_1 são dados por estes parâmetros, ou seja

$$Y_0 = \bar{X}_L \quad (9)$$

$$Y_1 = \bar{X}_H \quad (10)$$

Pode-se mostrar que

$$\frac{\gamma}{\gamma+1} \leq \frac{\bar{X}}{\bar{X}_H} \leq 1 \quad (11)$$

onde $\gamma = (M-\beta)/\beta$. De (11) observa-se que a faixa de variação de \bar{X} / \bar{X}_H é $1/(\gamma+1)$. Se esta faixa for dividida em Q intervalos, pode-se escrever \bar{X} / \bar{X}_H da seguinte forma

$$\frac{\bar{X}}{\bar{X}_H} = \frac{\gamma}{\gamma+1} + n \frac{1}{\gamma+1} \frac{1}{Q} \quad (12)$$

onde $0 \leq n \leq Q$, $n \in \mathbb{Z}$. De (12) pode-se escrever

$$n = \frac{\frac{\bar{X}}{\bar{X}_H} - \frac{\gamma}{\gamma+1}}{\frac{1}{(\gamma+1)Q}} \quad (13)$$

No esquema proposto em [6], transmite-se o plano de bits e a informação paralela composta por \bar{X} e n. Para cada bloco de imagem é calculado o valor de n usando (13), arredondando-se o resultado para o inteiro mais próximo. Pode-se mostrar que a partir destes parâmetros, obtém-se na recepção \bar{X}_H e \bar{X}_L .

A fim de reduzir a taxa de bits deste algoritmo, utiliza-se uma designação de códigos para n de comprimento variável. Verifica-se empiricamente que a incidência de blocos em que n assume valores altos ($n = M, M-1, M-2$) é grande, e por outro lado, são raros os blocos em que o valor de n é pequeno. Por esta razão, a utilização de um código por entropia (código de Huffman por exemplo) torna-se uma opção atrativa para reduzir a taxa de bits.

Vale ressaltar que os parâmetros σ e α calculados da imagem apresentam também uma distribuição de valores concentrada [9], o que, a princípio, favorece a utilização de códigos com comprimento variável para redução da taxa de bits. No entanto, ao aplicar quantização escalar não uniforme para estes parâmetros, a distribuição dos valores de σ e α é "espalhada", inviabilizando esta alternativa para redução da taxa de bits.

3. QUANTIZAÇÃO DA INFORMAÇÃO PARALELA

Em algoritmos BTC a técnica utilizada para a quantização da informação paralela é fator determinante no desempenho do algoritmo no que se refere à qualidade da imagem reconstruída e à taxa de bits resultante.

A quantização escalar apresenta a vantagem de ser simples a sua implementação, exigindo pouco esforço computacional. Em contrapartida, aloca uma porção

considerável da taxa de bits total para a codificação da informação paralela. A quantidade de bits necessária para codificar a informação paralela pode ser reduzida se for explorada a correlação entre blocos de uma imagem.

Uma alternativa [1],[5] consiste em codificar simultaneamente os dois parâmetros da informação paralela, utilizando-se um quantizador vetorial bi-dimensional apropriado. Neste caso, o par de parâmetros é tratado como um vetor bi-dimensional.

Outra proposta é codificar o par de parâmetros da informação paralela separadamente[2]. A matriz formada pelos valores de \bar{X} é dividida em blocos de tamanho $M_X = m_X \times m_X$ e quantizadas por vetores M_X -dimensional com $2^{B_X^V}$ níveis de quantização. Por sua vez, a matriz de valores do 2º parâmetro é particionada em blocos de tamanho $N^* = m^* \times m^*$, sendo estes blocos quantizados por vetores M^* -dimensional utilizando-se 2^{B^*} níveis. Portanto, neste caso, $B_X^V + B^*$ bits são suficientes para representar os blocos das matrizes de informação paralela, o que resulta em uma taxa de bits de $(B_X^V + B^*)/M(M_X + M^*)$.

Neste trabalho optou-se por comparar os métodos de quantização escalar e quantização vetorial separada, deixando de lado o método de quantização vetorial conjunta, uma vez que foi demonstrado em [7] que este último apresenta um desempenho inferior à quantização dos parâmetros em separado.

4. RESULTADOS DE SIMULAÇÃO

Nas simulações em computador foram utilizadas nove imagens monocromáticas. Seis imagens (Girly, Barby, Hursley, Hotel, Airplane, Goldy) foram utilizadas como seqüência de treinamento para projeto dos quantizadores e três (Lena, Baboon Peppers) foram empregadas como imagens de teste. Os dicionários contendo os vetores-código utilizados na quantização vetorial e quantização escalar foram gerados pelo emprego repetitivo da técnica de duplicação seguida do algoritmo LBG [8].

Neste trabalho utilizou-se como medida de desempenho a razão *signal de pico-ruído*, definida por

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\frac{1}{L} \sum_{i=1}^L (X_i - X_i^r)^2} \quad [\text{dB}] \quad (14)$$

onde L é o número total de pixels da imagem e X_i e X_i^r são valores dos pixels das imagens original e reconstruída, respectivamente.

A dimensão do vetor utilizada na quantização vetorial foi fixada em quatro para ambos os parâmetros \bar{X} e σ, α ou n. Ou seja, $M_X = M^* = 4$. A taxa de bits R_s foi variada entre 0,17 bit/pixel e 0,33 bit/pixel.

Na implementação da quantização escalar foram utilizados 3 a 8 bits para \bar{X} , enquanto que para os parâmetros σ e α utilizou-se 3 e 4 bits, resultando em uma taxa R_s que variou entre 0,38 e 0,75 bit/pixel. Demonstra-se em [9] que o emprego de uma quantidade de bits maior do que quatro para a codificação de σ ou α é desnecessário, uma vez que praticamente o desempenho não é alterado. Para a codificação de n, utilizou-se um código de bits de comprimento variável aplicado diretamente aos valores inteiros gerados a partir da equação (13).

A Tabela I mostra os resultados obtidos para a quantização escalar nos métodos $\{\bar{X}, \sigma\}$ e $\{\bar{X}, \alpha\}$, onde B_X^e e B^e são as quantidades de bits utilizadas para a quantização de \bar{X} e σ ou α . A Tabela II se refere à quantização escalar no método $\{\bar{X}, n\}$. Na Tabela II, B_X^e representa o número de bits utilizado para a quantização de \bar{X} e B^e a quantidade de bits média resultante da codificação do parâmetro n por comprimento variável. Os desempenhos dos codificadores BTC usando o esquema de quantização vetorial para codificação da informação paralela são apresentados na Tabela III.

Da Tabela I observa-se que, para toda a faixa de R_s empregada, o desempenho do método $\{\bar{X}, \alpha\}$ é apenas ligeiramente superior ao desempenho de $\{\bar{X}, \sigma\}$. No entanto, comparando-se os valores das Tabelas I e II, pode-se notar que a diferença de desempenho entre o método $\{\bar{X}, n\}$ e os demais é significativa. A Figura 1 mostra claramente esta diferença para a imagem de teste "Lena". A curva relativa ao método $\{\bar{X}, n\}$ parece transladada em relação às curvas dos demais métodos, mostrando que é possível conseguir um desempenho neste método comparável ao obtido nos outros esquemas, com uma taxa de bits inferior em pelo menos 0,05 bit/pixel. Este resultado se deve à codificação de comprimento variável implementada no método $\{\bar{X}, n\}$.

A Tabela III mostra que os desempenhos dos três algoritmos BTC, utilizando o esquema de quantização vetorial para a codificação da informação paralela, são praticamente idênticos. A Figura 2 demonstra isto claramente, apresentando o desempenho dos métodos aplicados para a codificação da imagem de teste "Lena". Ou seja, a vantagem em termos de desempenho apresentada pelo método $\{\bar{X}, n\}$ na quantização escalar da informação paralela, não é observada quando se opta pela quantização vetorial. Neste caso, a alternativa recomendada para codificação BTC utilizando-se o esquema de quantização vetorial é o método $\{\bar{X}, \alpha\}$, uma vez que demanda um menor esforço computacional para o cálculo do parâmetro α .

Por fim, das Tabelas I, II e III, pode-se observar que o esquema de quantização vetorial é a melhor opção quando se deseja a codificação de imagens com uma taxa de bits reduzida, para um nível de desempenho aceitável. Da Tabela III, verifica-se por exemplo que a imagem "Peppers" pode ser reconstruída com PSNR de 31,244 dB e uma taxa de bits de apenas 0,266 bit/pixel para a informação paralela. Optando-se pela

quantização escalar, com a melhor alternativa possível (método $\{\bar{X}, n\}$), esta mesma imagem deverá ser codificada com uma taxa de bits 68% maior para que se atinja o mesmo nível de desempenho. Vale ressaltar, no entanto, que a quantização

escalar é mais fácil de ser implementada, demanda um menor esforço computacional, e ainda, atinge níveis de desempenho não alcançados, na prática, pela quantização vetorial.

Tabela I
Desempenho dos algoritmos BTC usando quantização escalar de \bar{X} , σ e α

R_s	B^e	B_*^e	"Lena"		"Baboon"		"Peppers"	
			$\{\bar{X}, \sigma\}$	$\{\bar{X}, \alpha\}$	$\{\bar{X}, \sigma\}$	$\{\bar{X}, \alpha\}$	$\{\bar{X}, \sigma\}$	$\{\bar{X}, \alpha\}$
0,375	3	3	26,160	26,192	24,069	24,128	26,849	26,883
0,438	3	4	26,207	26,237	24,096	24,204	26,892	26,937
0,500	4	4	30,413	30,494	25,935	26,101	31,153	31,274
0,563	5	4	32,052	32,170	26,627	26,627	33,199	33,395
0,625	6	4	32,366	32,534	26,568	26,761	33,822	34,050
0,688	7	4	32,657	32,793	26,598	26,792	33,980	34,217
0,750	8	4	32,706	32,854	26,610	26,807	34,044	34,284

Tabela II
Desempenho dos algoritmos BTC usando quantização escalar de \bar{X} e n .

B_X^e	"Lena"			"Baboon"			"Peppers"		
	R_s	B_*^e	$\{\bar{X}, n\}$	R_s	B_*^e	$\{\bar{X}, n\}$	R_s	B_*^e	$\{\bar{X}, n\}$
3	0,321	2,139	26,104	0,343	2,481	24,041	0,322	2,150	26,764
3	0,385	3,155	26,217	0,401	3,416	24,184	0,366	2,860	26,913
4	0,447	3,155	30,532	0,464	3,416	26,156	0,429	2,860	31,294
6	0,509	2,139	32,272	0,530	2,481	26,592	0,509	2,150	33,392
7	0,571	2,139	32,388	0,593	2,481	26,624	0,572	2,150	33,538
8	0,634	2,139	32,434	0,655	2,481	26,637	0,634	2,150	33,595
8	0,697	3,155	32,906	0,714	3,416	26,887	0,679	2,860	34,353

Tabela III

Desempenho dos algoritmos BTC usando quantização vetorial de \bar{X} , σ , α e n .

R_s	B_X^v	B_n^v	"Lena"			"Baboon"			"Peppers"		
			$\{\bar{X}, \sigma\}$	$\{\bar{X}, \alpha\}$	$\{\bar{X}, n\}$	$\{\bar{X}, \sigma\}$	$\{\bar{X}, \alpha\}$	$\{\bar{X}, n\}$	$\{\bar{X}, \sigma\}$	$\{\bar{X}, \alpha\}$	$\{\bar{X}, n\}$
0,172	7	4	27,968	28,014	27,970	24,081	24,182	24,076	28,782	28,801	28,683
0,188	7	5	28,109	28,135	28,118	24,184	24,314	24,225	28,970	29,040	28,896
0,203	8	5	28,917	28,950	28,951	24,651	24,796	24,694	29,824	29,910	29,753
0,219	9	5	29,466	29,504	29,503	24,990	25,147	25,054	30,384	30,481	30,306
0,234	9	6	29,589	29,668	29,640	25,092	25,250	25,156	30,540	30,602	30,499
0,250	10	6	30,016	30,101	30,086	25,301	25,466	25,374	31,050	31,120	31,024
0,266	10	7	30,102	30,184	30,165	25,347	25,555	25,458	31,148	31,244	31,178
0,281	11	7	30,488	30,579	30,548	25,519	25,735	25,643	31,540	31,645	31,580
0,297	12	7	30,764	30,866	30,863	25,643	25,866	25,774	31,876	31,989	31,936
0,313	13	7	31,033	31,127	31,129	25,729	25,957	25,865	32,134	32,254	32,203
0,328	14	7	31,198	31,296	31,309	25,794	26,024	25,935	32,300	32,425	32,381

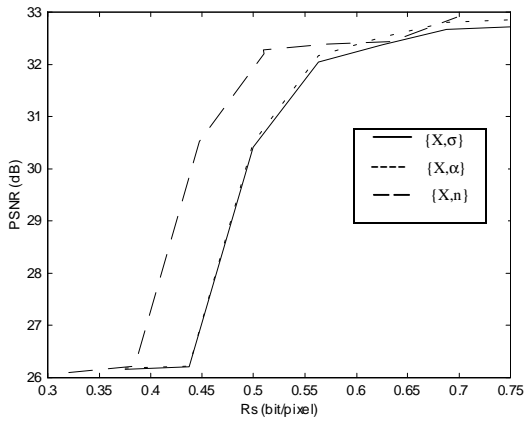


Figura 1: Desempenho dos algoritmos BTC usando quantização escalar para a imagem "Lena".

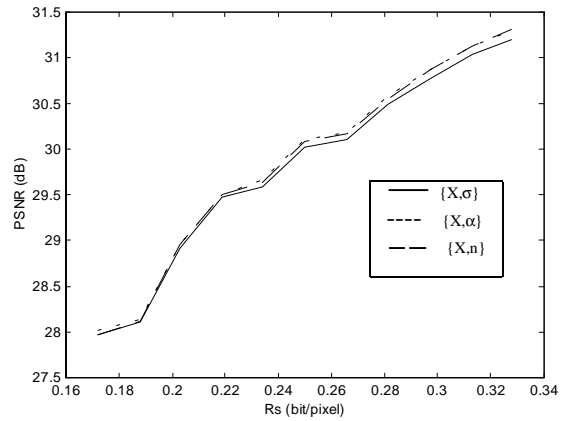


Figura 2: Desempenho dos algoritmos BTC usando quantização vetorial para a imagem "Lena".

5. CONCLUSÕES

Este trabalho realizou uma comparação entre as principais técnicas de codificação BTC de imagens, considerando o emprego de dois esquemas de quantização aplicados à informação paralela. Quando se utiliza quantização escalar, o método que emprega o parâmetro n como informação paralela apresenta menor taxa de bits que os demais, devido à possibilidade de emprego de codificação por entropia. Na quantização vetorial, os resultados de simulação mostram que os métodos BTC considerados neste trabalho têm desempenho praticamente equivalente no que se refere à PSNR. Neste caso, recomenda-se o emprego do método que utiliza o momento absoluto central como informação paralela, pois é a alternativa que apresenta menor esforço computacional. Finalmente, comparando-se os esquemas de quantização escalar e vetorial, observa-se que este último reduz significativamente a taxa de bits da codificação, e ainda mantém um nível de desempenho de PSNR comparável ao obtido na quantização escalar. Em contrapartida, apresenta maior complexidade para implementação.

REFERÊNCIAS

- [1] E.J. Delp and O. R. Mitchell, "Image Compression using block truncation coding", *IEEE Trans. Commun.* Vol. COM-27, pp 1335-1342, Setembro 1979.
- [2] M.D.Lema e O. R. Mitchell, "Absolute Moment block truncation coding and its application to colour images", *IEEE Trans. Commun.*, vol. COM-32, pp 1148-1157, Outubro 1984.
- [3] O. R. Mitchell e E.J. Delp, "Multilevel graphics representation using block truncation coding", *Proc IEEE*, vol. 68, pp. 868-873, Julho 1980.
- [4] V.R. Udpikar e J.P. Raina, "Modified algorithm for block truncation coding of monochrome images", *Electronics Letters*, vol. 21, pp 900-902, Setembro 1985.
- [5] V.R. Udpikar e J.P. Raina, "BTC image coding using vector quantization", *IEEE Trans. Commun.*, vol. COM-35, pp.352-356, Março 1987.
- [6] Y.V. Ramana e C. Eswaran, "New bit rate reduction techniques for block truncation coding", *IEEE Tran. Commun.*, Vol. 44 pp 1247-1250, Outubro 1996.
- [7] L. V. Oliveira e A. Alcaim, "Vector quantization of the side information in BTC image coding", *Proc. ICCS/ISITA '92, Singapore*, pp 345-349, 1992.
- [8] Y. Linde, A. Buzo, e R.M. Gray, "An algorithm for vector quantizer design", *IEEE Trans. Commun.* , Vol. COM-28, pp 84-95, Janeiro 1980.
- [9] L. V. Oliveira e A. Alcaim, "Sensibilidade dos parâmetros estatísticos em codificadores de imagem BTC", *Anais do XI Simpósio da Sociedade Brasileira de Telecomunicações*, Natal, RN, pp 660-664, 1993.