

# Compressão de Imagens Através de Quantização Vetorial Classificada e Predição de Médias

JOSÉ EDUARDO ALVES DE MOURA<sup>1</sup>  
ABRAHAM ALCAIM<sup>1</sup>

<sup>1</sup>CETUC-PUC/Rio  
Rua Marques de S. Vicente, 225  
22453-900 Rio de Janeiro, RJ, Brasil  
alcaim@cetuc.puc-rio.br

**Abstract.** In this paper we present an image compression technique which employs classified vector quantization and mean prediction. We describe a perceptual block classification algorithm with reduced computational complexity which determines the class for each image block. The block is then encoded with a vector quantizer designed specifically for that class. Mean prediction is used to improve the performance of smooth areas of the image at low bit rates. In this case a residual vector quantizer is employed. Run-length coding and Huffman coding are used to reduce the bit rate required to encode the side information.

## 1 Introdução

Quantização vetorial é uma técnica que tem sido amplamente utilizada para codificação digital de imagens com o objetivo de transmissão ou armazenamento [Gersho, Gray (1992)], [Nasrabadi, King (1988)]. Isso se deve principalmente à sua capacidade de codificação a baixas taxas de bits utilizando blocos de tamanhos relativamente pequenos. Além disso, a estrutura do decodificador é bastante simples, o que torna o método atraente para aplicações que envolvem um codificador e vários decodificadores (como videotexto, por exemplo).

A entrada de um quantizador vetorial de imagens é um vetor  $\mathbf{X}$  que representa um bloco de pixels da imagem. O codificador calcula a distorção  $d(\mathbf{X}, \mathbf{Y}_i)$  entre o vetor de entrada  $\mathbf{X}$  e cada um dos vetores-código  $\mathbf{Y}_i$ ,  $i = 1, 2, \dots, N$ , que compõem um dicionário de códigos  $\mathbf{C}$ . O quantizador ótimo é aquele que escolhe o vetor-código que resulta na menor distorção. O índice associado a esse vetor-código é, então, digitalizado. Um algoritmo muito utilizado para projetar o dicionário de códigos  $\mathbf{C}$  a partir de uma sequência de treinamento, e que converge para um dicionário ótimo num sentido local, é o algoritmo LBG [Linde, Buzo, Gray (1980)].

Dois problemas importantes associados a um quantizador vetorial são a alta complexidade do processo de busca no dicionário de códigos e a baixa qualidade da representação de bordas. A complexidade computacional do codificador cresce exponencialmente com a taxa de bits e com o tamanho do bloco. O problema mais sério, porém, é a degradação de bordas. Esse problema está associado ao fato

de que os critérios de distorção em geral utilizados, como o erro médio quadrático (EMQ), não garantem a integridade das bordas na imagem reconstruída. Esse critério não tem a propriedade de preservação de bordas. Portanto, um bloco de bordas pode não ser codificado com um vetor-código de bordas. Além disso, devido à sua probabilidade de ocorrência relativamente baixa, esses blocos tenderão a não estarem bem representados no dicionário de códigos. Uma vez que bordas representam uma informação visual muito importante, resulta que a qualidade global da imagem reconstruída poderá ser significativamente deteriorada.

Quantização vetorial classificada (QVC) [Ramamurthi, Gersho (1986)] tem demonstrado ser uma técnica de grande utilidade para tratar os dois problemas discutidos acima. Em QVC a imagem é dividida em blocos e cada bloco é primeiramente classificado por um algoritmo de classificação como pertencente a uma dentre várias classes pré-definidas. Em seguida, cada bloco é codificado usando um dicionário de códigos projetado especificamente para sua classe. Cabe à regra de distorção o papel de realisar um melhor ajuste entre blocos que já possuem características semelhantes. Como são utilizados dicionários menores, a complexidade do processo de busca pelo melhor vetor-código pode ser reduzida de forma significativa. Além disso, como blocos que possuem bordas são codificados utilizando dicionários cujos vetores também possuem bordas, a integridade das bordas é melhor preservada. QVC é capaz de preservar características importantes de percepção e ainda manter um critério de distorção simples como

EMQ. Com o objetivo de melhorar a qualidade de blocos que representam áreas suaves da imagem em baixas taxas de bits, será utilizado aqui um procedimento de quantização vetorial residual desses blocos. O resíduo é obtido subtraindo-se de cada pixel o valor predito da média do bloco.

O algoritmo de classificação, que tem a vantagem de apresentar uma baixa complexidade computacional, é descrito na Seção 2 deste trabalho. A Seção 3 compara o algoritmo aqui utilizado com aquele descrito em [Ramamurthi, Gersho (1986)]. Os dois esquemas são comparados em termos de desempenho e complexidade computacional para diferentes alocações de bits por classe. A Seção 4 apresenta o método de predição de médias e quantização vetorial residual usado para regiões da imagem com características mais uniformes. Os resultados de desempenho são apresentados na Seção 5, juntamente com as técnicas de corrida de classes e código de Huffman utilizadas para redução da taxa de bits para representar a informação paralela (classe do bloco). As principais conclusões do trabalho são fornecidas na Seção 6.

## 2 Descrição do Classificador

Inicialmente a imagem é dividida em blocos de tamanho  $4 \times 4$ . Cada um desses blocos é, então, classificado como pertencente a uma dentre 11 classes distintas que serão descritas a seguir.

Para blocos de dimensões pequenas, como as utilizadas nesse trabalho, uma **borda** pode ser definida como uma linha reta através da qual ocorre uma mudança abrupta de intensidade. Consideramos aqui 8 classes de bordas, de acordo com suas orientações e polaridades:

- $\pm 0rd$ : bordas horizontais
- $\pm \pi/2rd$ : bordas verticais
- $\pm \pi/4$  e  $\pm 3\pi/4rd$ : bordas diagonais

A polaridade está relacionada com o sentido da variação através da borda: de uma maior intensidade para uma menor e vice-versa. Um bloco é considerado **uniforme** se não possuir variação significativa de intensidade, **meia-faixa** se possuir gradiente moderado mas nenhuma borda e **misto** se possuir alto gradiente porém sem uma borda dominante bem definida. Como será visto posteriormente, a maior parte dos blocos pertence à classe **meia-faixa**.

A mesma definição de classes aqui considerada foi empregada no algoritmo proposto em [Ramamurthi, Gersho (1986)]. Esse algoritmo utiliza dois passos, realizando sempre o mesmo número de

operações por bloco, independentemente da classe a que pertence o bloco. Ele é baseado em duas matrizes que representam os gradientes (modificados) horizontais e verticais. Já no algoritmo aqui considerado explora-se a distribuição dos blocos através das classes como forma de reduzir a complexidade. Procura-se realizar um menor número de operações para os blocos cujas classes têm maior probabilidade de ocorrência.

No primeiro passo, o classificador decide por uma classe uniforme com base em um simples gradiente normalizado entre a máxima ( $MAX$ ) e mínima ( $MIN$ ) intensidade dos pixels pertencentes ao bloco:

$$G_u = \frac{2(MAX - MIN)}{MAX + MIN} \quad (1)$$

Foi utilizado o gradiente normalizado porque a visão humana é mais sensível às variações *normalizadas* de intensidade. Se  $G_u < T_u$  então o bloco é **uniforme**. Um valor adequado para a constante  $T_u$  de forma a permitir uma boa separação entre as classes uniforme e meia-faixa é 0,05.

Se o bloco não for uniforme ele apresenta uma variação de intensidade de moderada a alta. Como blocos de bordas estão incluídos nessa categoria, uma medida natural desta variação é o gradiente  $G$  calculado a partir de diferenças orientadas entre as médias de regiões adjacentes pertencentes ao bloco [Chen, Bovik (1990)]. Sejam  $m_e$  e  $m_d$  as médias das metades esquerda e direita do bloco, respectivamente, e sejam  $m_s$  e  $m_i$  as médias das metades superior e inferior do bloco, respectivamente.  $G$  é definido por

$$G = \sqrt{G_x^2 + G_y^2} \quad (2)$$

onde

$$G_x = \frac{2(m_e - m_d)}{m_e + m_d} \quad (3)$$

e

$$G_y = \frac{2(m_s - m_i)}{m_s + m_i} \quad (4)$$

Se  $G < T_m$  o bloco não possui bordas bem definidas porém apresenta uma variação de intensidade moderada já que  $G_u \geq T_u$ . Neste caso o bloco é classificado como **meia-faixa**.

Se  $G \geq T_m$  o bloco contém borda ou pertence à classe mista. Neste ponto define-se um limiar de detecção de borda  $T_b$  tal que se  $G_u > T_b$  então a variação entre os valores máximo e mínimo dos pixels pertencentes ao bloco é significativa e o bloco é considerado como um bloco de **borda**. Caso contrário, emprega-se o algoritmo proposto em [Ramamurthi, Gersho (1986)] para detetar a classe **mista**. Neste caso, se o bloco não for classificado como misto ele

será considerado um bloco de **borda**. Valores adequados para os limiares foram  $T_b = 0, 8$  e

$$T_m = \begin{cases} 8/MED & \text{se } MED < 30 \\ 0,15 & \text{caso contrario} \end{cases} \quad (5)$$

onde  $MED$  é a intensidade média do bloco. Pode-se verificar experimentalmente que quando  $MED$  é baixa melhores resultados são obtidos com um valor de  $T_m$  que dependa de  $MED$  [Ramamurthi, Gersho (1986)].

A orientação e a direção das bordas são determinadas como se segue. Primeiramente note que para blocos  $4 \times 4$  há 3 possibilidades para cada borda horizontal ou vertical e 4 para cada borda diagonal ( $\pi/4$  ou  $3\pi/4$ ), resultando em um total de 14 tipos de bordas. Sejam  $a_i$  e  $b_i$  duas regiões do bloco separadas por uma borda do tipo  $i$  e sejam  $m_{a_i}$  e  $m_{b_i}$ , respectivamente, as intensidades médias dessas regiões. A **orientação** da borda será aquela para a qual a magnitude do gradiente normalizado

$$G_i = \frac{2(m_{a_i} - m_{b_i})}{m_{a_i} + m_{b_i}} \quad (6)$$

é máxima. A **polaridade** é dada pelo sinal de  $G_i$ .

### 3 Desempenho e Complexidade do Classificador

Foram projetados diversos conjuntos de dicionários de códigos para as 11 classes de atributos visuais. Eles foram gerados através do uso repetitivo de particionamento seguido do algoritmo LBG [Linde, Buzo, Gray (1980)]. Um total de 15 imagens foram utilizadas na seqüência de treinamento. A imagem teste Lena ( $512 \times 512$  pixels) encontra-se fora da seqüência de treinamento. A medida objetiva de desempenho considerada é a razão pico-ruído:

$$RPR = 10 \log_{10} \frac{255^2}{\frac{1}{L} \sum_{k=1}^L (s_k - \hat{s}_k)^2} \quad (7)$$

onde  $L$  é o número de pixels da imagem e  $s_k$  e  $\hat{s}_k$  são os pixels das imagens original e reconstruída, respectivamente. As alocações de vetores por classe consideradas são mostradas na Tabela 1 e os desempenhos versus taxas de bits são mostrados na Tabela 2 para o esquema proposto e o descrito em [Ramamurthi, Gersho (1986)], aqui identificado como "Referência". Detalhes das imagens original e reconstruídas pelos dois esquemas são mostradas na Fig. 1. Os resultados mostram que os esquemas são comparáveis em termos de desempenho e taxa de bits, com uma pequena vantagem para o esquema proposto.

Em termos de complexidade computacional, o algoritmo proposto apresenta uma vantagem significativa sobre o da referencia. O esforço computacional é reduzido em aproximadamente 40 %. A complexidade computacional foi medida aqui com base na probabilidade de ocorrência de cada classe (estimada a partir da seqüência de treinamento), no número de operações necessárias para identificar cada classe, e em um conjunto de coeficientes de ponderação associados a cada tipo de operação [Dall'Agnol, Alcaim, De Marca (1994)].

Classe	No. de Vetores/Classe				
	1	2	3	4	5
Diagonal	256	512	512	512	256
Vert./Horiz.	128	256	256	256	128
Meia-faixa	512	512	512	1024	1024
Uniforme	64	64	64	64	64
Mista	128	128	512	512	512
No. da Alocação	1	2	3	4	5

Tabela 1: Alocações de vetores por classe.

No. da Alocação	1	2	3	4	5
Taxa (bpp) Ref.	0,77	0,78	0,80	0,83	0,82
RPR (dB) Ref.	30,1	30,4	30,8	30,9	30,6
Taxa (bpp) Prop.	0,77	0,82	0,80	0,83	0,82
RPR (dB) Prop.	30,6	31,0	31,1	31,2	30,8

Tabela 2: Desempenhos versus taxas.

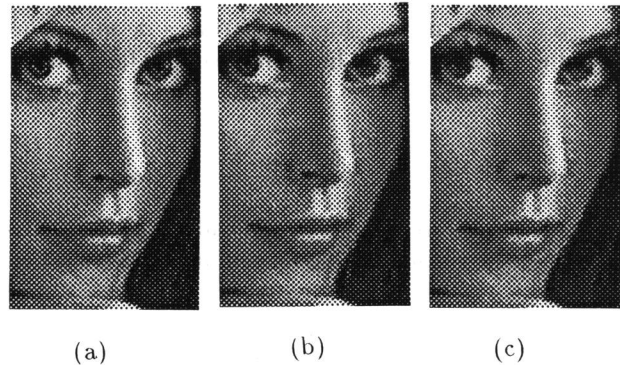


Figura 1: Detalhes das imagens original (a) e reconstruídas pelos esquemas referência (b) e proposto (c).

As probabilidades de ocorrências das 11 classes de atributos visuais são dadas na Tabela 3. Note que a maioria dos blocos pertencem às classes meia-faixa e uniforme, para as quais a complexidade do classificador proposto é a mais baixa.

Classe	Probabilidade
$-\pi/4$	0.030
$+\pi/4$	0.033
$-3\pi/4$	0.032
$+3\pi/4$	0.030
$-\pi/2$	0.032
$+\pi/2$	0.033
$-0$	0.035
$+0$	0.035
Mista	0.030
Uniforme	0.080
Meia-Faixa	0.630

Tabela 3: Probabilidades das classes.

#### 4 Predição de Médias e Quantização Vetorial Residual

A taxa de bits total do codificador é fortemente dependente da taxa associada à classe meia-faixa. Além disso, uma representação inadequada dessa classe, através de quantizadores vetoriais com baixo número de vetores por dicionário, tende a introduzir efeitos de blocagem que deterioram a qualidade visual da imagem reconstruída [Lo, Cham (1994)].

Com o objetivo de reduzir o número de níveis do dicionário sem introduzir degradações significativas na imagem reconstruída, Nasrabadi e Feng (1990) propuseram uma técnica de predição de médias seguida de quantização vetorial residual. Basicamente o que é feito é prever a média do bloco a partir de valores de pixels codificados em uma vizinhança do bloco, subtrair essa média de cada pixel do bloco e codificar essa diferença através de um quantizador vetorial. A predição da média é dada por

$$m = \frac{1}{9} \sum_{k=1}^9 X_k \quad (8)$$

onde os  $X_k$ 's são os pixels codificados nas fronteiras horizontal e vertical do bloco, conforme ilustra a Fig. 2.

Como o decodificador dispõe dos blocos codificados, não há necessidade de bits adicionais para representar a média. O decodificador pode determiná-la a partir dos blocos decodificados e somá-la ao vetor do dicionário de modo a obter o bloco reconstruído.

Como os blocos pertencentes à classe meia-faixa são os que ocorrem com maior frequência, e como para esses blocos foram necessários dicionários de maior tamanho, o procedimento descrito foi empregado apenas para esses blocos. Com isso buscou-se reduzir a taxa de bits sem deterioração significativa da qualidade visual para taxas mais baixas.



Figura 2: Ilustração dos pixels codificados utilizados para predição de médias.

#### 5 Resultados de Desempenho

A estrutura do sistema de codificação aqui proposto incorpora ainda um procedimento de corrida de classes e codificação de Huffman [Huffman (1952)], de modo a reduzir a taxa de bits usada para representar a informação paralela. A corrida de classes explora a ocorrência de seqüências relativamente longas de blocos pertencentes às classes uniforme e meia-faixa, e o código de Huffman tira proveito da distribuição não-uniforme de probabilidades das classes.

Para cada seqüência de blocos consecutivos pertencentes à mesma classe (uniforme ou meia-faixa) utiliza-se, então, os seguintes parâmetros para representá-la:

- $C$ : código de comprimento variável (Huffman) indicando a classe
- $F$ : "flag" (1 ou 0) indicando se há ou não corrida
- $L$ : comprimento (número de blocos) da corrida
- $i_1, i_2, \dots, i_L$ : índices dos vetores obtidos do dicionário de códigos da classe

Para as outras classes são usados apenas os parâmetros  $C$  e  $i_1$ .

O diagrama em blocos do codificador é mostrado na Fig. 3. Como o decodificador consiste basicamente em associar os vetores apropriados aos índices

correspondentes, achou-se desnecessário incluir seu diagrama em blocos na figura.

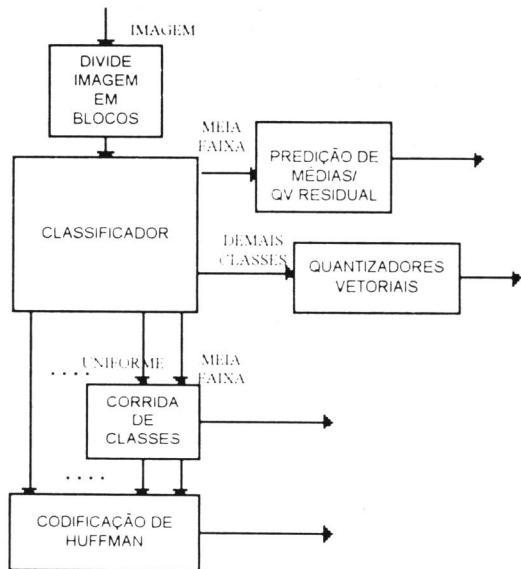


Figura 3: Diagrama em blocos do codificador.

Os desempenhos obtidos para novas alocações de vetores por classe, em que foram apenas reduzidos os tamanhos dos dicionários utilizados para a classe meia-faixa, são mostrados na Tabela 4. As imagens reconstruídas quando se utiliza as alocações de vetores 2 e 4 (Tabela 1) sem predição de médias e as correspondentes alocações quando se reduz a taxa usada para a classe meia-faixa (2a. e 4a. colunas da Tabela 4) e emprega-se predição de médias, são mostradas nas Figs. 4 e 5. Embora o desempenho objetivo tenha decaído, o emprego de predição de médias permitiu reduzir a taxa de bits sem deterioração perceptível da qualidade visual.

Classe	No. de Vetores/Classe			
Diagonal	256	512	512	512
Vert./Horiz.	128	256	256	256
Meia-faixa	32	64	128	256
Uniforme	64	64	64	64
Mista	128	128	512	512
Taxa (bpp)	0.47	0.52	0.57	0.60
RPR (dB)	29.6	30.4	30.6	30.7

Tabela 4: Resultados de desempenho.



(a)



(b)

Figura 4: Imagens reconstruídas: (a) sem predição de médias à taxa de 0.82 bpp; (b) com predição de médias à taxa de 0.52 bpp.



(a)



(b)

Figura 5: Imagens reconstruídas: (a) sem predição de médias à taxa de 0,83 bpp; (b) com predição de médias à taxa de 0,57 bpp.

## 6 Conclusões

Este artigo apresentou uma técnica de compressão digital de imagens que utiliza quantização vetorial classificada em 11 categorias. O algoritmo de classificação utilizado fornece uma redução de complexidade de cerca de 40 % em relação ao método proposto em [Ramamurthi, Gersho (1986)], mantendo taxa de bits e desempenho comparáveis. Com o objetivo de reduzir a taxa de bits usada para identificação da classe, o esquema de codificação incorpora um procedimento de corrida de classes para as classes uniforme e meia-faixa, e codificação de Huffman para todas as classes.

Para regiões da imagem com características mais suaves, e que ocorram com alta probabilidade, o método de codificação apresentado neste trabalho faz uso de predição de médias e quantização vetorial residual. Esse procedimento permite reduzir a taxa de bits total em mais de 30 % sem degradação significativa da qualidade visual da imagem reconstruída.

## 7 Referências

- D. Chen and A. C. Bovik, Visual pattern image coding, *IEEE Transactions on Communications* **38** (1990) 2137–2146.
- S. L. Q. Dall’Agnol, A. Alcaim and J. R. B. De Marca, Performance of vector quantizers for VSELP coders in noisy channels, *European Transactions on Telecommunications and Related Technologies* **5** (1994) 553–563.
- A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression* Kluwer Academic Publishers, 1992.
- N. A. Huffman, A method for the construction of minimum redundancy codes, *Proceedings of the IRE* **40** (1952) 1098–1101.
- Y. Linde, A. Buzo and R. M. Gray, An algorithm for vector quantizer design, *IEEE Transactions on Communications* **28** (1980) 84–95.
- N. T. Lo and W. K. Cham, New predictive classified vector quantisation scheme for image compression, *Electronics Letters* **30** (1994) 1280–1282.
- N. M. Nasrabadi and R. A. King, Image coding using vector quantization: a review, *IEEE Transactions on Communications* **36** (1988) 957–971.
- N. M. Nasrabadi and Y. Feng, Image compression using address-vector quantization, *IEEE Transactions on Communications* **38** (1990) 2166–2173.
- B. Ramamurthi and A. Gersho, Classified vector quantization of images, *IEEE Transactions on Communications* **34** (1986) 1105–1115.