New Method to Reduce the Size of Codebook in Vector Quantization of Images

Sahar K. Ahmed
College of Engineering
Mosul University

Received on: 7/3/2004 **Accepted on: 13/12/2004**

الملخص

إن طريقة التكميم الاتجاهي المهمة في كبس الصور تولد دليلاً رمزياً يجب أن يكون متوفراً في أثناء عملية كبس الصور و استرجاعها مما يؤدي الى الحاجة لإرسال الدليل إلى جهة الاستقبال عند إرسال الصورة المكبوسة والى خزن هذا الدليل في الحاسبة التي تحتوي على صورة مكبوسة باستخدام طريقة التكميم الاتجاهي. ومن هنا ظهرت أهمية تقليص حجم الدليل لزيادة الكفاءة الكلية لعملية الكبس.

في هذا البحث تم التوصل الى وسيلة لتقليل حجم الدليل الرمزي المتولد في طريقة التكميم الاتجاهي وذلك باستخدام ترتيب خاص لكلمات الدليل الرمزي(codewords) . ومن خلال تطبيق (Huffman coding)على الاختلاف بين الكلمات المتتابعة مما ينتج عنه التقليل في حجم الدليل.

ABSTRACT

The vector quantization method for image compression inherently requires the generation of a codebook which has to be made available for both the encoding and decoding processes. That necessitates the attachment of this codebook when a compressed image is stored or sent. For the purpose of improving the overall efficiency of the vector quantization method, the need arose for improving a means for the reduction of the codebook size.

In this paper, a new method for vector quantization is presented by which the suggested algorithm reduces the size of the codebook generated in vector quantization. This reduction is performed by sorting the codewords of the codebook then the differences between adjacent codewords are computed. Huffman coding (lossless compression) is performed on the differences in order to reduce the size of the codebbook.

1. Introduction

A fundamental goal of data compression is to reduce the bit rate for transmission or data storage while maintaining an acceptable fidelity or image quality [3]. Image data compression has been pushed to the forefront of the image processing field. This is largely a result of the rapid growth in computer power, the corresponding growth in the multimedia market, and the advent of the world wide web, which makes the Internet easily accessible for everyone. Additionally, the advances in video technology, including high-definition television, are creating a demand for a new, better, and faster image compression technology [5]. Another area for the application of efficient coding is where pictures are stored in a database, such as archiving medical images, multispectral images, finger prints, and drawing [3].

There are two primary types of image compression methods. The first type is called lossless methods because no data are lost, and the original image can be retrieved exactly from the compressed data. The second type is called lossy methods because they allow a loss in the actual image data, so the original uncompressed image cannot be retrieved exactly from the compressed file [5].

Image data compression using vector quantization(lossy compression) has received a considerable attention because of its simplicity and adaptability [4]. The first empirical design scheme is suggested by Linde,Buzo, and Gray[2], and thus named LBG algorithm, This scheme is a generalization of Lioyd PCM design technique[1].Vector quantization(VQ) requires the input image to be processed as vectors and finds the best or closest match, based on some distortion criterion, from its stored codebook. The address of the best match is then transmitted to the decoder. The decoder accesses an entry from an identical codebook, thus obtaining the reconstructed vector. Data compression is achieved in this process because the transmission of the address requires fewer bits than transmitting the vector itself[4]. A review of vector quantization techniques used for image coding is presented by Nasrabadi and King[3].

The codebook is needed in the encoding and decoding process. In this paper the reduction of the codebook size is achieved by taking the differences between the closest codewords. It is noted that small differences occurred more frequently than large differences, so Huffman coding is applied on the differences.

The Huffman code, developed by D.Huffman in 1952, is a minimum length code. This means that, given the statistical distribution of the gray levels(histogram), the Huffman algorithm will generate a code that is as close as possible to the minimum bound, the entropy. This method results in a variable length code, a property useful in reducing the size of the codebook. A small length code should be given to the differences occurring frequently.

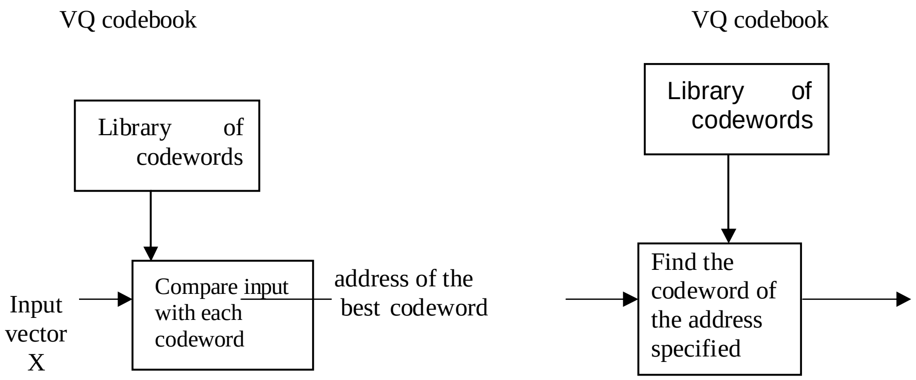
2. Vector Quantization

A vector quantizer can be defined as a mapping Q of k-dimensional Euclidean space into a finite subset Y of R^k. A mapping Q is defined as follows:

$$Q: R^k \longrightarrow Y$$

Where $Y = (\hat{X}_i , i = 1,2,..., N)$ is the set of reproduction vectors (codewords) and N the number of vectors in Y[9]. This finite set Y is called a VQ codebook or VQ table. By choosing the size of codebook Y, we can control the transmission rate of a VQ coding process. Our goal is to select an optimal codebook Y of size N that results in the lowest possible distortion among all possible codebooks of the same size [1]. VQ requires the input image to be processed as vectors or blocks of image pixels[4]. The encoder views the input vector X and generates the address of the codeword specified by Q(X). The address is then transmitted to the decoder. The decoder uses this address to generate the reconstructed vector $\hat{X}$ see figure(1). If a distortion measured $d(X,\hat{X})$ which represents the penalty or cost associated with reproducing vectors X by $\hat{X}$ is defined, then the best mapping Q is the one which minimized $d(X,\hat{X})$[3]. The Most Common measure of distortion between X and $\hat{X}$ is the sequared Euclidean distance, given by[8].

$$d(X, \hat{X})=\|X- \hat{X}\|^2=\sum_{j=0}^{k-1}(X(j)-\hat{X}(j))^2$$


VQ codebook
Library of codewords
Input vector X
Compare input with each codeword
address of the best codeword
VQ codebook
Library of codewords
Find the codeword of the address specified

(a)encoder (b)decoder

Figure (1): VQ of image (a) encoder, (b) decoder

2.1 Codebook Design

The goal of designing an optimal vector quantizer is to obtain a codebook consisting of N codewords, such that it minimizes the expected distortion.

The LBG algorithm will split the image into the same size parts and forms training vectors $X_j(k)$ as:[$X_j(1)$ $X_j(2)$ $X_j(3)$…….. $X_j(k)$]

wherej=1,2,3,……,n; n=number of the image parts.

k=vector dimension;(k=1,2,3,………,16).

This algorithm generates a codebook Y containing N codewords as

$$\hat{X}_i(k) = [\hat{X}_i(1)\,\hat{X}_i(2)\hat{X}_i(3)..........\hat{X}_i(k)]$$

Where i=1,2,3,…………….,N; N=number of codewords.

The algorithm for LBG is given in [3] as follows:

1- Let N=number of levels(codewords); distortion threshold $\epsilon >_{=} 0$. Assume an initial N level reproduction alphabet (codebook) Y_0, and a training sequence (X_j:j=0,1,--------,n-1), and m=number of iteration, set to zero.

2- Given Y_m=($\hat{X}_i$; i=1,--------,N), find the minimum distortion partition $p(Y_m)$=(S_i;i=1,--------,N)of the training sequence :

$X_j \in S_i$ if $d(X_i,\hat{X}) \le d(X_i,\hat{X}_1)$ for all l. Compute the average distortion

$$D_m=D[(Y_m,\ p(Y_m))]=n^{-1} \sum_{j=0,\hat{X}eY_m}^{n-1} \min\ d(X_j,\hat{X}).$$

Where D_m: average distortion at iteration m.

S_i: partition i of the training vectors (i=1,2,------,N).

3- if (D_{m-1}-D_m)/ D_m<= ϵ, stop the iteration with Y_m as the final codebook; otherwise continue.

4- Find the optimal codebook

$$\hat{X}(P(Y_m)) = (\hat{X}(s_i), i = 1,2,----------, N)\ for P(Y_m) where$$

$$\hat{X}(s_i) = \frac{1}{\| s_i \|} \sum_{j:Xj\in Si} X_j$$

5- Set $Y_{m+1} = \hat{X}(s_i)$, increment m to m+1, and go to (2).

In the above iteration algorithm, initial codewords must be assumed. As reported by Equitis, the performance of his algorithm is better than that of the LBG algorithm with a randomly selected initial codebook. As suggested by Equits, an even better codebook can be constructed by first using the Equitis algorithm to obtain the initial codebook and then using the LBG algorithm to refine the codebook.

At the beginning of the Equitis NN algorithm, all the training vectors are viewed as initial codewords. Then, the two nearest codewords are merged and replaced by a new codeword which is a weighted average of the two merged codewords. Hence, the number of codewords after merging is reduced by one. This process is repeated until the desired number of codewords is reached. If all the possible pairs are to be compared at each merging step, the total computation becomes too numerous [1].

In this work, to ease the computation, the initial codebook is specified as follows: at the beginning all the training vectors are viewed as initial codewords. These codewords are sorted depending on the squared Euclidean distance criteria . Then, the two adjacent codewords are merged and replaced by the average of the two codewords, so at the end the number of codewords is divided by two. The merging process is repeated until the desired size of the codebook is reached.

After the design of the codebook is completed, the enhancement of the codebook size is obtained as follows:

1.Sort the codewords of the codebook depending on the squared Euclidian distance criteria(2.2). Hence, we obtain the successive codewords as close as possible.

2.Compute the differences between successive codewords, It is found that these differences are frequently of small values.

3.Apply Huffman coding(variable length code) on the differences obtained in step2.

3. Results and Discussion

The VQ of images are performed on different images of size 512*512 pixels and 256 gray levels(8 bit per pixel). The initial codebook is specified as suggested in this paper, sorting all training vectors then merging every successive vector. The merging process is repeated until the desired size of the codebook is reached.The size of codebook is 1024 codewords. The training vectors are obtained by scanning the image with a fixed size window(4*4).

Figures 2,3, and 4 represent the original and decoded images of Lena, Girl and Car, respectively.

The fidelity criteria used to measure the amount of error in the reconstructed (decompressed) image is the peak signal-to-noise ratio which is defined in an N×N decompressed image as follows(Umbaugh, 1998):

$$\text{PSNR}= \quad 10\ \log_{10} \frac{(L-1)^2}{\frac{1}{N^2}\sum_{r=0}^{N-1}\sum_{c=0}^{N-1}[\hat{\mathrm{I}}(\mathrm{r},\ \mathrm{c}) - \mathrm{I}(\mathrm{r},\ \mathrm{c})]^2}$$

Where I(r,c)=the original image
Î(r,c)=the decompressed image
L=the number of gray levels(where L=256)
r=row, c=column

The PSNRs of the decoded images are tabulated in table(1). The results obtained give a good quality of the decoded images, the bit per pixel value is 0.625.

Table 1: PSNR of the decoded images

Image	PSNR
Lena	35.12
Girl	38.10
Car	39.26

The codebook would be stored or transmitted as well as the codebook addresses. We obtain the reduction in the codebook size by ordering the codewords, then taking the differences between successive ones. After that Huffman coding (Lossless compression) is performed. The number of bits required for each codeword pixel element are tabulated in table (2).

Table 2: Bits per codeword pixel element

Image	Bits per codeword pixel element(LBG)	Bits per codeword pixel element (enhanced codebook)	Percentage of reduction
Lena	8	6.45	19.375%
Girl	8	5.7	28.75%
Car	8	5.6	30%

The decoder accesses the codewords specified by the addresses. Each codeword pixel element represents a pixel value of the decoded image. We have 256 gray level images, so 8 bits per codeword pixel element are needed. The bit per codeword pixel element obtained in the enhanced codebook, shown in table(2), is less than that of the LBG codebook.

4. Conclusion

Vector quantization is one of the most important methods in the field of image compression, this method is applied onto three images and the result provides a high compression ratio with a good image quality. In this paper, it had been shown that the size of the codebook can be reduced by sorting the codeword in the codebook depending on the squared Euclidean distance criteria, then the differences between successive codewords are computed. After that, Huffman coding is applied on the differences.

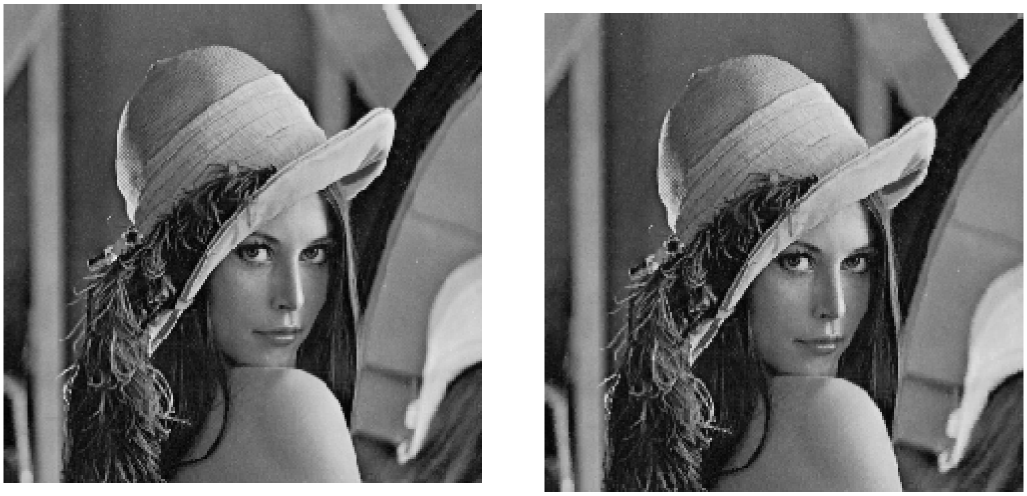


Figure 2: (a) Source Lena Image (b)DecodedLena Image

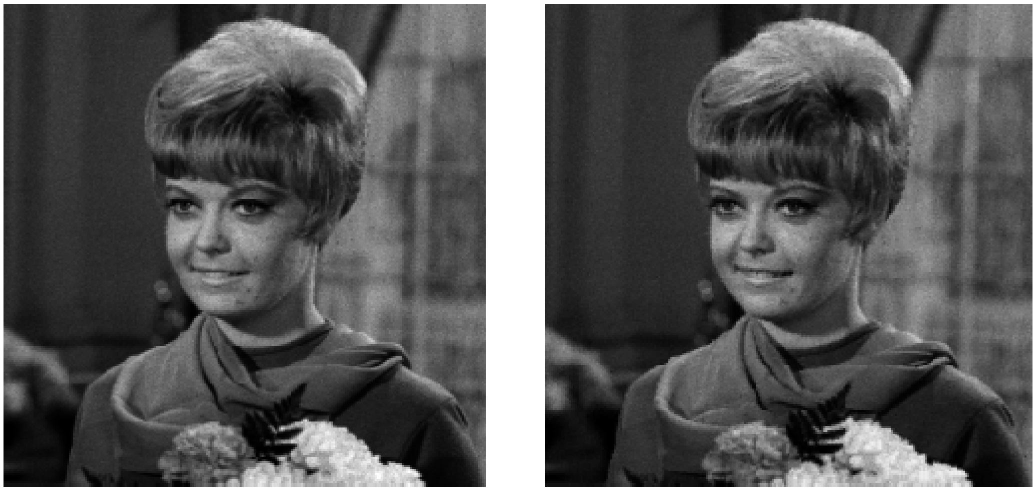


Figure 3: (a) Source Girl Image (b) Decoded Girl Image

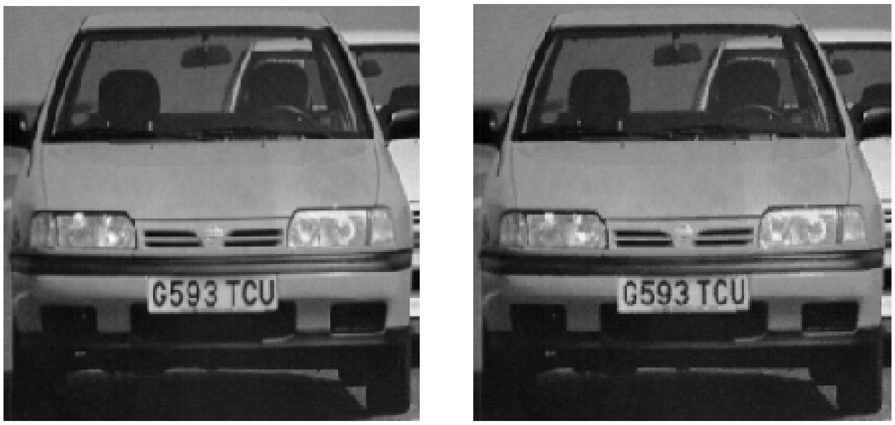


Figure 4: (a) Source Car Image (b) Decoded Car Image

REFERENCES

[1] Hang, H. & Haskell, B. G., April 1988, "**Interpolative Vector Quantization of Color Image**", *IEEE Transactions on Communications, Vol. 36, No. 4*, PP. 465-470.

[2] Linde, Y.; Buzo, A.; Gray, R. M., January, 1980., "**An Algorithm for Vector Quantization Design**", *IEEE Transaction on Communications, Vol. Com-28,No. 1*, PP. 84-95.

[3] Nasrabadi, N. M. & King, R. A., August 1988, "**Image Coding Using Vector Quantization: A Review**", *IEEE Transaction on Communications, Vol. 30, No. 8*, PP 957-971.

[4] Sitaram, V. S.; Huang C.; Israelsen, P. D., November 1994, "**Efficient Codebooks for Vector Quantization Image Compression with an Adaptive Tree Search algorithm**", *IEEE Transaction on Communications, Vol. 42, No. 11*, PP. 3027-3033.

[5] Umbaugh ,S. E.,1998, "**Computer Vision and Image Processing**", *Prentice Hall Inc.*

[6] Walker, N. P.; Eglen S. J.; Lawrence B. A., 1994, "**Image Compression Using Neural Networks**,".*GEC Journal of Research, Vol. 11, No. 2*, PP. 66-75.

[7] Zeger, K.; Vaisey, J., Gersho, A., February 1992, "**Globally Optimal Vector Quantization Design by Stochastic Relaxation**", *IEEE Transaction on Signal Processing, Vol. 40, No. 2*, PP. 310-322.

[8] LU Zhe-Ming, XU Dian-Guo, July 2003, "**Fast Codeword Search Algorithm for Image Vector Quantization Based on Ordered Hadamard Transform**",*IEICE Trans. Inf.&Syst., Vol. E86-D, No. 7*, PP. 1318-1320.

[9] Pantsaena, N.; Sangworasil, M.; Nantajiwakornchai C. and Phanprasit T.,2002, "**Image Compression using Vector Quantization"**, *WWW.Kmutt.ac.th/itc 2002/CD/pdf/18-07-45/TA2-p/1.pdf.*