International Journal of Advanced Technology in Engineering and Science
Vol. No.4, Issue No. 05, May 2016
www.ijates.com
ijates
ISSN 2348 - 7550

A COMPARISON ON VECTOR QUANTISATION AND SMVQ

Simi N[1], Teena Rajan[2]

[1]*M.tech, schlor, Dept. of Electronics & Communication*

[2]*Mar Baselios College of Engineering & Technology*

Trivandrum, (India)

ABSTRACT

Side match vector quantization (SMVQ) is an image compression scheme that reduces the redundancy of a digital image. Some data hiding methods utilize SMVQ to embed secret data in an SMVQ-decompressed image, and reconstruct the original SMVQ code after the secret data have been extracted. It is an advanced form of Vector Quantisation(VQ). In this paper we are comparing the performance of VQ and SMVQ methods using LBG algorithm and a random code book design algorithm by evaluating the peak signal to ratio(PSNR) of compressed images.

Keywords: Image compression,PSNR, side match vector quantization (SMVQ),vector quantisation (VQ).

I. INTRODUCTION

The advancement in the information and internet technology enable the people to transmit and share digital contents conveniently. In such a scenario ensuring communication efficiency and save network bandwidth, compression techniques can be implemented on digital content. In many applications, most digital content, especially digital images and videos are converted into the compressed forms for transmission.

Vector quantization (VQ) is an effective compression scheme of digital images for the purpose of transmission and storage [1-5]. The major advantages of VQ are that the compression rate is very high and its design is very simple. In general,VQ consists of three phases: (1) codebook generation phase, (2) the encoding phase and (3) the decoding phase. At first, a codebook K, which is composed of the most representative code words, must be constructed. Then K will be employed in both the encoding phase and the decoding phase. Generating a perfect codebook from a large amount of training set is a critical work in VQ. Many codebook generation algorithms have been proposed, and the most famous one is the LBG algorithm that was presented by Linde, Buzo and Gray in 1980 [6]. Basically, the LBG algorithm is an iterative algorithm that splits the training sets and updates the codebook iteratively.

As an improved version of VQ, SMVQ[7-10] has been introduced which make use of original VQ codebook and a sub-codebook to perform compression of digital images, in which both are used to generate the index values, excluding the blocks in the leftmost column and the topmost row.In this paper a comparison and effectiveness of both VQ and SMVQ have been analyzed and studied based on LBG algorithm and a random code book design algorithm. The rest of the paper is organized as follows. Section IIdescribes the method of SMVQ and basic principle of SMVQ, Section III describes LBG algorithm and design of random code book algorithm. Experimental results and analysis are given in Section IV, and Section V concludes the paper.

II. CONCEPT OF SMVQ

To improve the compression bit rate, side match VQ was proposed by Kim in 1992 [7]. Kim assumes the pixels in the top row in the current block are correlated closely with those in the bottom row in the upper block, and the pixels in the first column in the current block are correlated closely with those in the right column in the left block, pixels in the fourth column in the current block are correlated closely with those in the left column in the right block, and that the pixels in the bottom row in the current block are correlated closely with those in the top row in the lower block.

Based on this assumption, Kim used Side-Match approach to design SMVQ, and successfully reduces the blocking effect by using local edge information and provides better visual quality and compression ratio than VQ does. To perform SMVQ, a super codebook is required to encode the blocks in the first row and the first column, and a state codebook is required to encode the rest of the blocks. The state codebook is a subset of the super codebook. Consider the 4x4 image blocks shown in Figure 1, where U and L are image blocks reconstructed by the traditional VQ, and X is the current processing image block.

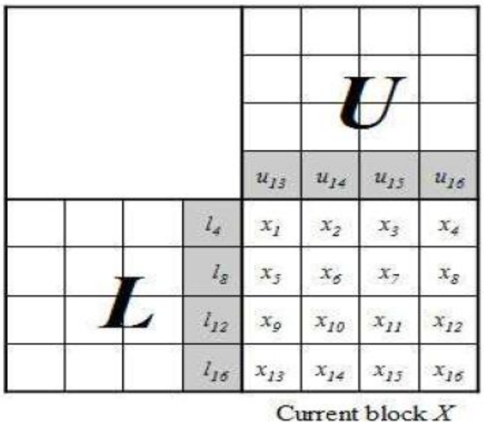

Fig 1:Current ,upper ,left and neighbouring blocks.

The image blocks in the first row and first column are denoted as seed blocks, and they are encoded by using the traditional VQ technique with super codebook Y sized N. The rest of the blocks are defined as residual blocks, and they are used in SMVQ encoding. Each residual block is encoded with the assistance of upper, left neighboring block of the current processing block X.

To encode the residual block, the SMVQ encoding phase consists of four steps. First, the boundary value of block X is predicted using the values of neighboring values of block U and block L, such as :

$$x_1 = \frac{u_{13}+l_4}{2} \qquad (1)$$

$$x_2 = u_{14} \qquad (2)$$

$$x_3 = u_{15} \quad (3)$$

$$x_4 = u_{16} \quad (4)$$

$$x_5 = l_8 \quad (5)$$

$$x_9 = l_{12} \quad (6)$$

$$x_{13} = l_{16} \quad (7)$$

Then, the predicted value is used to look up codewords in super codebook Y to generate the corresponding state code book. A state codebook contains M codewords that are selected from super codebook Y . These M codewords have the minimum side-match distortion, when compared with the gray areas in Fig. 1. Here, the minimum side-match distortion is calculated by using Eq. (8). After the state codebook has been generated, the current residual block is encoded by the best-match codeword of the current residual block, which is obtained by searching the state codebook instead of the super codebook.

$$d(x, y_i) = \sum_{j=1}^{k} (x_j - y_{i,j})^2 \quad (8)$$

In the SMVQ decoding phase, first, the indices in the first row and first column are decoded by VQ with the super codebook as was done in the encoding phase. To reconstruct residual blocks, the previously decoded upper and left neighboring blocks are used to generate a state codebook that contains M codewords with the minimum side-match distortion for the current block. Then, the state codebook is searched fully to find the mapping codeword required for the received index to recover its block. After all received indices have been processed; the reconstruction of the original image is obtained.

III. LBG ALGORITHM

LBG algorithm is like a K-means clustering algorithm which takes a set of input vectors S = { ∈ | = 1, 2 . . . n} as input and generates a representative subset of vectors C = { ∈ | j = 1, 2. . . K} with a user specified K << n as output according to the similarity measure. For the application of Vector Quantization (VQ), K = 256 or 512 are commonly used.

1.Input training vectors S = { ∈ | = 1, 2, · · ·, n}.

2. Initiate a codebook C = { ∈ | = 1, 2, · · ·, K}.

3. Set = 0 and let k = 0.

4. Classify the training vectors into K clusters according to ∈

5. Update cluster centers , = 1, 2, · · ·, .

6. Set $k \leftarrow k$ - and compute the distortion

7. If (D_{k-1} −)/Dk > (a small no:), repeat steps 4 -6.

8. Output the codebook C = { ∈ | j = 1, 2, · · ·, K}.

The convergence of LBG algorithm depends on the initial codebook C, the distortion , and the threshold , in implementation, we need to provide a maximum number of iterations to guarantee the convergence.

International Journal of Advanced Technology in Engineering and Science
Vol. No.4, Issue No. 05, May 2016
www.ijates.com
ijates
ISSN 2348 - 7550

A random code book algorithm was generated iteratively with size N=256 in multiples of 4.

IV. EXPERIMENTAL RESULTS

Experiments were conducted on a group of gray-level images to compare the effectiveness of SMVQ over VQ. In the experiment, the sizes of the divided non-overlapping image blocks were 4 × 4, i.e., n = 4. Accordingly, the code book size N was N=256. Six standard, 512 × 512 test images, i.e., Lena, Peppers and Baboon, are shown in Figure 2.

(a) (b) (c)

Fig 2: Input Images (a) Lena.jpg (b) Peppers_Gray.jpg (c) Baboon.jpg

Both SMVQ and VQ compression has been done in these test input images using LBG algorithm and Random code book design algorithm. The performance and effectiveness is evaluated using Peak signal to noise ratio(PSNR). Fig 3 gives the output of images compressed using random code book generation algorithm.

(a) (b) (c)

Fig.3. Output Image using Random code book generation (a) Lena.jpg (b) Peppers_Gray.jpg (c) Baboon.jpg

Figure 4 shows the compression results of images using LBG algorithm.

(a) (b) (c)

Fig.4. Output Image using LBG Algorithm (a) Lena.jpg (b) Peppers_Gray.jpg (c) Baboon.jpg

Peak signal-to-noise ratio (PSNR) was utilized to measure the visual quality of the decompressed images, see Eq. (9).

ijates
ISSN 2348 - 7550

$$PSNR = 10 \times \log_{10} \frac{255^2 \times M \times N}{\sum_{x=1}^{M}\sum_{y=1}^{N}[I(x,y)-I_d(x,y)]^2} \quad (9)$$

M and *N* are the height and the width of the images, respectively; $I(x, y)$ and $I_d(x, y)$ are the pixel values atcoordinate (x, y) of the original uncompressed image Iand the decompressed image .Compression ratio was also evaluated see Equ. 10.

$$CR = \frac{8\times N}{L} \quad (10)$$

where L is the length of compressed codes.

TABLE I: IMAGE QUALITY ASSESSMENT

Input Image	PSNR		
	Random Code book algorithm	LBG Algorithm	SMVQ
Lena.jpg	22.8167	32.2630	32.246
Peppers.jpg	21.0595	31.0369	32.012
Baboon.jpg	23.1334	27.5998	27.6008

TABLE II: COMPRESSION RATIO

Input Image	CR	
	VQ	SMVQ
Lena.jpg	16	20.08
Peppers.jpg	16	19.04
Baboon.jpg	15	18.45

Comparing Fig .3 and 4 shows that compression using LBG algorithm yields better visual quality of images. Moreover comparing VQ and SMVQ, it shows that SMVQ considers the correlation between the neighboring pixels which increase the compression ratio than VQ.

V. CONCLUSION

A comparison on VQ and SMVQ based on LBG algorithm and random code book generation algorithm has been discussed. Results shows that compression using LBG algorithm yields better quality images than Random code generation algorithm. Moreover the visual quality of both VQ and SMVQ compressed images are slightly similar,SMVQ provides a better compression method than VQ,since they consider the correlation between neighbouring pixels. Here instead of considering all the pixels,each pixel is represented using an index table, thus improving the compression ratio.

REFERENCES

[1] R. M. Gray, Vector quantization,IEEE ASSP Mag., vol. 1, no. 2, pp. 4-29, 1984.

[2] W. J. Wang, C. T. Huang, and S. J. Wang, "VQ applications in steganographic data hiding upon multimedia images," IEEE Syst. J.,vol. 5, no. 4, pp. 528–537, Dec. 2011.

[3] Y. C. Hu, "High-capacity image hiding scheme based on vector quantization," Pattern Recognit., vol. 39, no. 9, pp. 1715–1724, 2006.

[4] Y. P. Hsieh, C. C. Chang, and L. J. Liu, "A two-codebook combination and three-phase block matching based image-hiding scheme with high embedding capacity," Pattern Recognit., vol. 41, no. 10, pp. 3104–3113, 2008.

[5] C. H. Yang and Y. C. Lin, "Fractal curves to improve the reversible data embedding for VQ-indexes based on locally adaptive coding," J. Vis. Commun. Image Represent., vol. 21, no. 4, pp. 334–342, 2010.

[6] Y. Linde, A. Buzo, and R. M. Gray, An algorithm for vector quantizer design, IEEE Trans. On Communs., vol. 28, no. 1, pp. 84-95, 1980.

[7] T. Kim, "Side match and overlap match vector quantizers for images,"*IEEE Trans. Image Process.*, vol. 1, no. 2, pp. 170–185, Apr.1992.

[8] C. C. Chang, W. L. Tai, and C. C. Lin, "A reversible data hiding scheme based on side match vector quantization," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 10, pp. 1301–1308, Oct. 2006.

[9] C. C. Chen and C. C. Chang, "High capacity SMVQ-based hiding scheme using adaptive index," Signal Process., vol. 90, no. 7,pp. 2141–2149, 2010.

[10] P. Tsai, "Histogram-based reversible data hiding for vector quantisation compressed images," IET Image Process., vol. 3, no. 2, pp. 100–114,2009.

[11] W. B. Pennebaker and J. L. Mitchell, The JPEG Still ImageDataCompression Standard. New York, NY, USA: Reinhold, 1993.

[12] A. Gersho and R. M. Gray, Vector Quantization and Signal Compression. Norwell, MA, USA: Kluwer, 1992.

[13] C. Qin, C. C Chang and Yi-Ping Chiu, "A novel joint data-hiding and compression scheme based on SMVQ and image inpainting," IEEE Trans. Image Process., vol. 23, no. 3, March. 2014.

[14] S. C. Shie and L. T. Chen, Image compression based on side- match VQ and SOC, ProcceedingsofDigital Image Computing: Techniques and Applications, pp. 369-373, 2009.