

Image To Document Conversion Using Difference Vector Quantization And Zig-Zag Technique

¹J.Sheik Mohamed Meera

Master of Engineering

Department of Computer Science & Engineering

SSN College of Engineering, Chennai

²S.Lakshmi Priya

Assistant Professor

Department of Computer Science & Engineering

SSN College of Engineering, Chennai

Abstract

Image compression involves the task of minimizing the image size without degrading the quality of image. It reduces the memory space required for storing the large scale image databases. Better compression rate can be achieved by applying difference vector quantization(DVQ) technique on the image. But this technique is not suited to compress images in real time applications like email. Since email, multitype of datatypes such as jpeg,png, tiff etc are used and more over in the existing work, the mean value, index value and codeword have been used. In order to reduce the problem of using multiple types of datatype usage in email and the transmission problems while transferring them in the existing work,this paper proposes a technique for image compression by combining difference vector quantization with zigzag technique. Image is first transformed into blocks and DVQ is applied on each block. Then the processed blocks are trained using Linde-Buzo-Gray(LBG) algorithm to generate a codebook. The image is converted into a list of documents by rearranging the index value using zig-zag technique. The compressed image is reconstructed by matching the index value of the codeword with the codebook. Experimental results shows that the proposed image compression technique reconstructs images quicker and reduces the storage space.

Keywords: DVQ(Difference Vector Quantization), Code book, Zig-zag

I INTRODUCTION

Due to the recent advances in digital media technology, image compression has become a hot area of research. Images

are widely distributed on internet. When transmitting an image, it is usually compressed to reduce the image size and increase the transmission speed.

Several techniques are available to compress an image. DVQ technique [1],[2],[3] is a lossy compression techniques. The basic idea of DVQ is to select some benchmark images which are used to train a codebook. Codebook contains a collection of codewords. Codewords are obtained from the input image by applying DVQ. For each codeword of the input image the corresponding matching index value from the codebook is obtained. Original image is reconstructed using the index value of the codeword. The original image with 512X512 pixels is taken and it is splitted into 4096 blocks. Each block size is set to 4X4. The resulting benchmark image codebook has 256X16 codewords. Each codeword has its respective index value, which can be represented in 8bits. After encoding the original image, the codebook contains 4096 codeword in the form of 8 bit indices.

In DVQ compression technique, for each image block $[d_{ij}]$, its mean value $[m_i]$ is found. Then the mean value is subtracted from the original block value ($d_{ij} = b_{ij} - m_i$).The resulting block is encoded using the index value which points to a codeword in the codebook.

If the pre-trained codebook has 256 codewords, then each index value can be represented in 8 bits. After the original image being encoded into an index table, it shall contain 4096x8 bit indices. In other words, the original 4096x16 pixels image blocks can be compressed into 4096X8 bit indices. Hence the compression rate is 16x8 bits. Higher compression rates can be achieved using DVQ, but the decompressed

image obtained is of poor quality. This is because image reconstruction is performed by using similar codewords selected from codebook. Hence the decompressed image quality is essentially affected by the pre-trained codebook.

II. RELATED WORK

The two basic tasks of vector quantization image compression are: the design of the codebook and the search for the best codeword for each block [4],[5].

A. VECTOR QUANTIZATION

Vector Quantization (VQ) [6] is a lossy data compression method based on the principle of Block Coding. It is a fixed length algorithm. The design of the vector quantizer is considered to be a challenging problem due to the need for multi-dimensional integration .

Linde, Buzo and Gray (LBG) [7] proposed a VQ design algorithm based on training sequence. The use of the training sequence bypasses the need for multidimensional integration. Each quantizer codeword represents a single sample of the source output. By taking longer sequences of input samples, it is possible to extract the structure in the source coder output. Even when the input is random, encoding sequences of samples instead of encoding individual samples separately provides a more efficient code. But the major drawback of LBG is that the utility of each codeword in the codebook is low.

Fuzzy clustering method [8],[9] improves the quality of selected images vector for the codebook training. Shen and Huang [10] proposed an adaptive image compression method that uses the difference value between the original image and the VQ-decompressed image to raise the VQ-decompressed image quality. Ra and Kim proposed a fast mean-distance-ordered partial codebook search algorithm [11]. They used squared mean distance (SMD) to filter false candidate code-words. Patane and Russo proposed a clustering algorithm called enhanced LBG (ELBG). They used the concept of utility of a codeword to overcome the local optimal problem of LBG.

In this paper we have proposed a method for image compression which utilizes both difference vector quantization with zigzag technique. The paper is organized as follows. Section III gives a detailed description of the proposed method. Section IV provides experimental results and section V contains a summary.

III. METHOD

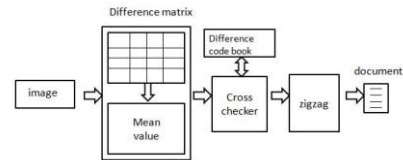


Figure 2.1:Compression

Figure 2.1 describes the compression process. The input image is split into a 4096 of blocks. Then each block is transformed into vector matrix. Mean value is calculated from the vector matrix. The original vector matrix is subtracted based on this mean value. A new vector matrix is derived from this process and it is called as difference vector quantization matrix. The difference codebook is defined with code words and code vectors. Then code word in the codebook will generate an equivalent index value. The code words of the input image are cross checked with the default difference codebook and then latter will be updated if necessary. For each codeword in the input image, a similar codeword from the codebook is found by calculating the Euclidean distance. The corresponding index value now represents that codeword. All the indices of all the codewords of the input image as obtained and written as a document using zigzag technique.

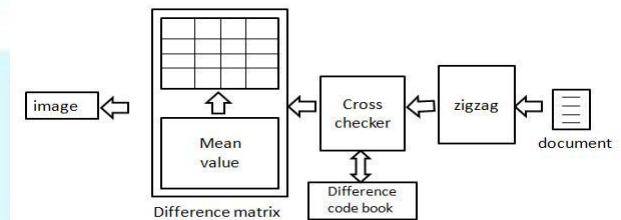


Figure 2.2:Decompression

Figure 2.2 describes the Decompression process of the document. The index value is compared with the code word in codebook. Then this code word is cross checked with the difference codebook and the mean value is added to produce the difference vector quantization matrix. This value is converted into blocks and the original image is again reconstructed.

A. ENCODING

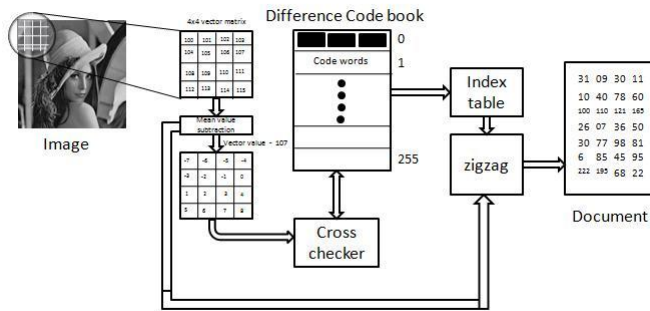


Figure 2.3:Encoding

Figure 2.3 describes the encoding process. First the image is splitted into blocks and the blocks are then converted into vector quantization matrix. Mean is calculated for each vector quantization matrix(4x4). Using that mean value the vector quantization matrix is subtracted and a new difference vector quantization matrix is generated. Initially a code book is constructed using the LBG algorithm. Cross checker will compare the difference vector value and the default difference code book and finally bench mark image codebook is updated. The index value of the corresponding code word available in the code book will be stored in the index table. The index values from index table are taken and using a zigzag algorithm the values are written in a document.

B. DECODING

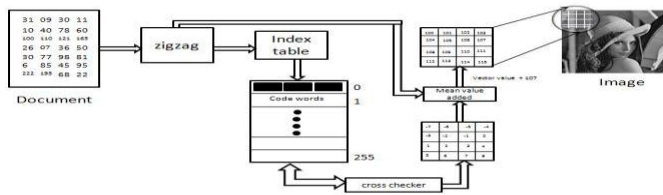


Figure 2.4:Decoding

Figure 2.4 describes the Decoding process In decoding , the document is read using the zigzag algorithm . The index values are cross checked with the default local difference code book and a difference quantization matrix is generated. A mean value is added with the difference matrix. The available vector quantization matrix is converted into blocks and the image is regenerated from the blocks.

C. LBG: Linde-Buzo-Gray ALGORITHM TO TRAIN THE CODEBOOK

LBG algorithm is like a k-means clustering algorithm which takes a set of input vectors $S = \{x_i \in R^d, i = 1; 2; \dots; n\}$ generates a representative subset of vectors $C = \{c_j \in R^d, j = 1; 2; \dots; K\}$ with a user specified K less than n as output according to the similarity measure. For the application of Difference Vector Quantization(DVQ), $d=16, K=256$ or 512 are commonly used.

- 1) Input training vectors $S = \{x_i \in R^d, i = 1; 2; \dots; n\}$ g.
- 2) Initiate a codebook $C = \{c_j \in R^d, j = 1; 2; \dots; K\}$ g.
- 3) Set $D_0 = 0$ and let $k = 0$.
- 4) Classify the n training vectors into K clusters according to $x_i \in S_q$ if

$$kx_i \cdot c_{qk_p} > kx_i \cdot c_{jk_p} \text{ for } j \neq q$$

- 5) Update cluster centers $c_j, j = 1; 2; \dots; K$ by

$$c_j = \frac{1}{j} \sum_{i \in S_j} x_i$$

- 6) Set $k = k + 1$ and compute the distortion,

$$D_k = \sum_{i=1}^n \|x_i - c_{j_k}\|^2$$

- 7) if $(D_{k-1} - D_k) = D_k > \epsilon$ (a small number), repeat steps

4 6. d

- 8) Output the codebook $C = \{c_j \in R^d, j = 1; 2; \dots; K\}$ g.

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

Difference Code book Design:

For building the code book, the LBG algorithm is used. The

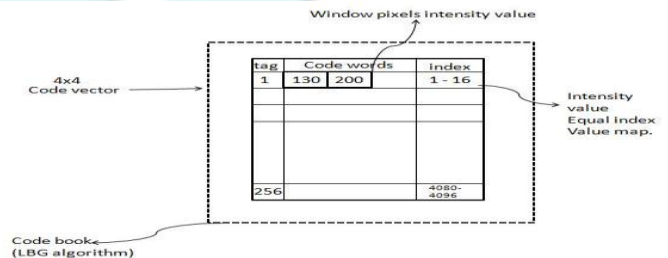


Figure 2.5: codebook design

input for the LBG code book is a training set of code vectors. The code book contents include tag, code words and index. Tag is used to represent the column size of code books row and column. The Window pixels intensity value is similar to code vector. The code vector equivalent index value is mapped. (eg: for a 512*512 image the 4*4 blocks 4096 index value are mapped).

IV. CODEBOOK TO DOCUMENT CONVERSION TECHNIQUE USING ZIGZAG TECHNIQUE

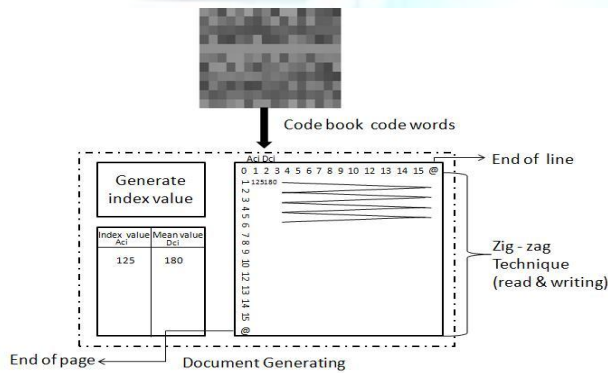
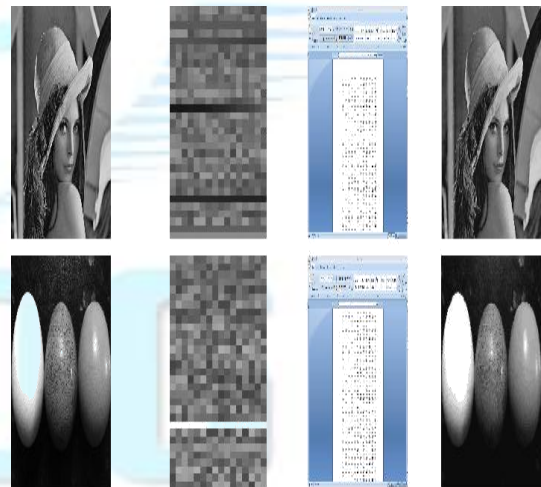
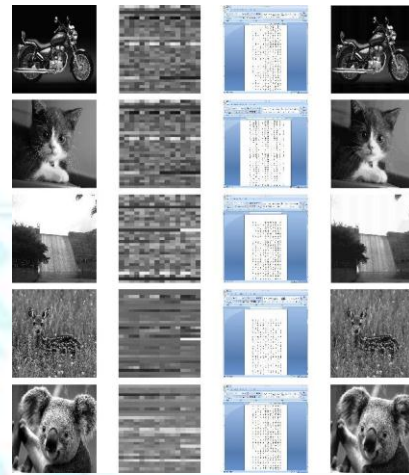


Figure 3: codebook to document

Figure 3 shows the index value of the code words that has been generated in the code book. The index value is stored in the table and the mean value used in the DVQ process that represents that block is also stored in the same table. The conventions used are AC_i for the index value and DC_i for the mean value. This AC_i and DC_i together forms a code word. Here we use vertical zig-zag technique to write the values present in the table. This writing is done during compression and reading is done during decompression. In this document the column of @ symbol signifies the end of the line and the row of @ symbol signifies the end of the page. During compression, the documents default properties such as fonttype, fontsize, zig-zag ordering are used to get and write the documents.

V. RESULTS OF IMAGE COMPRESSION



a) original image b) codebook c) document d) output

a) original image is the input image of size 512x512. b) codebook is obtained from the transformation of the original image. c) document is obtained from the transformation of codebook, document contains codeword values and the index value equal to the codeword and mean value. d) document is transformed to reconstructed image.

Peak signal to-noise ratio (PSNR) value is the destruction value that is found in the reconstructed image which is obtained from the original image and it is tabularized below

Image	Using VQ Technique	Using DVQ Technique
Bike	25.9843	24.4658
Cat	23.4568	22.8614
Dam	27.3256	26.7351
Deer	26.2456	26.3265
Kola	24.3913	23.7489
Lena	28.4828	29.6345
Three eggs	29.0504	28.2564

TABLE I: Image PSNR value

The above table I reveals PSNR values for different images obtained from reconstructed images using VQ technique and DVQ technique.

Image	Using VQ with zigzag Technique	Using DVQ with zigzag Technique
Bike	14 KB	16 KB
Cat	14.2 KB	16.3 KB
Dam	14.2 KB	16.2 KB
Deer	14.5 KB	16.8 KB
Kola	14.1 KB	16.5 KB
Lena	15 KB	17 KB
Three eggs	14.9 KB	16.7 KB

TABLE II: Document Size

The table II above reveals the size of the documents which is obtained from the codebook using VQ techniques and DVQ techniques.

VI. CONCLUSION

In this paper a simple image compression technique has been proposed. First, difference vector quantization is done on the image blocks. Then codebook is generated by training the

blocks using LBG algorithm. The images are compressed and converted into documents. Experiments have demonstrated that this approach yields better reconstructed image and reduces the storage space. The future work is to apply some encryption techniques during image compression to improve security features.

ACKNOWLEDGMENT

The authors would like to thank the reviewers for their valuable comments that helped to improve this paper.

REFERENCES

- [1] Jau-JiShen, Ya-Hsin LoA, "New Approach of Image Compression Based on Difference Vector Quantization Dalian", China October 14-October 2011 16 ISBN:78-0-7695-4517
- [2] Jian-JiunHsin-Hui chen, "Context-based adaptive zigzag scanning for image", Tainan 6-9 Nov. 2011 E-ISBN : 978-1-4577-1320-08
- [3] Ms. Asmita A. Bardekar, Mr. P.A. Tijare, "Implementation of LBG Algorithm for Image Compression", International Journal of Computer Trends and Technology volume 2 Issue 2- 2011.
- [4] Tzu-Chuen Lu, Ching-Yun Chang, "A Survey of VQ Codebook Generation", Journal of Information Hiding and Multimedia Signal Processing Volume 1, Number 3, July 2010.
- [5] Asmita A. Bardekar, P.A. Tijare, "Implementation of LBG Algorithm for Image Compression", International Journal of Computer Trends and Technology- volume 2, Issue 2- 2011.
- [6] P. Franti, "Genetic algorithm with deterministic crossover for vector quantization", Pattern Recognition Letters, 21 (1), 61-68, 2000.
- [7] Y. Linde, A. Buzo, "An Algorithm for Vector Quantizer Design", IEEE Transactions on Communications January 1980.
- [8] G. E. Tsekouras, "A fuzzy vector quantization approach to image compression", Applied Mathematics and Computation, vol. 167, no. 1, pp. 539-560, Aug. 2005.
- [9] W. Xu, A. K. Nandi "Novel fuzzy reinforced learning vector quantization algorithm and its application in image compression", IEEE Proceedings Vision, Image, and Signal Processing, vol. 150, no. 5, Oct. 2003.
- [10] J. J. Shen, H. C. Huang, "An adaptive image compression method based on vector quantization", International Conference on Pervasive Computing Signal Processing and Applications (PCSPA), no. 17-19, pp. 377-381, 2010.
- [11] S. W. Ra, J. K. Kim, "A Fast Mean-Distance-Ordered Partial Codebook Search Algorithm for Image Vector Quantization", IEEE Trans. Circuits and Systems - II: Analog and Digital Signal Processing, vol. 40, no. 9, pp. 576-579, 1993.