International Journal of Advanced Computer Technology (IJACT)

ISSN:2319-7900

IMAGE COMPRESSION WITH EFFICIENT CODEBOOK INITIALIZATION USING LBG ALGORITHM

Manoj Kumar, Poonam Saini,
Department of Computer Science & Engineering, National Institute of Technical Teachers' Training & Research, Chandigarh-160019, India, Tel.: +91 9411178729

Abstract

Vector quantization (VQ) has received a great attention in the field of multimedia data compression since last few decades because it has simple decoding structure and can provide high compression ratio. In general, algorithms of VQ codebook generation focus on solving two kinds of problem: (i) to determine the quantization regions and the code words that minimize the distortion error. (ii) to reduce the computational complexity of code words search for building the codebook. In this paper, a novel VQ codebook generation method based on the Linde-Buzo-Gray (LBG) is presented. VQ based image compression technique has three major steps namely (i) Codebook Design, (ii) VQ Encoding Process and (iii) VQ Decoding Process. The performance of VQ based image compression technique depends upon the constructed codebook. In the same sequence, widely used technique for VQ codebook design is the LBG algorithm. However the performance of the standard LBG algorithm is highly dependent on the choice of the initial codebook. In this paper, we have proposed a simple and very effective approach for image compression through LBG algorithm. The simulation results show that the proposed scheme is computationally efficient and gives expected performance.

Keywords: Image Compression, Codebook generation, LBG algorithm, Vector Quantization (VQ).

1. Introduction

Image data compression is concerned with reducing the number of bits required to represent an image or reducing bit rate for data set while maintaining an acceptable image quality [1-3]. There are two main application areas of image compression: Image transmission and archiving. The potential applications for image transmission are digital TV, Digital HDTV, video conferencing, facsimile transmission of printed material, graphics images, or the transmission of remote sensing images obtained from satellites and reconnaissance aircraft. Another area for the application of efficient coding is where pictures are stored in a database for archiving medical images, and in multimedia devices such as CD-I and 3DOS (Three Dimensional Operating System). The compression ratios required ranged from 8:1& 40:1, depending on the image modality. Although the technical constraints depend upon the application, the theoretical problem of data compression remains the same [4-10].

Digital images commonly contain lots of redundant information, and thus they are usually compressed to remove redundancy and minimize the storage space or transport bandwidth [11]. Two types of compression are possible: lossless and lossy. If the process of redundancy removing is reversible, i.e. the exact reconstruction of the original image can be achieved, it is called lossless image compression; otherwise, it is called lossy image compression. Scientific or legal considerations make lossy compression unacceptable for many high performance applications such as geophysics, telemetry, non-destructive evaluation, and medical imaging, which will still require lossless image compression [7, 9, 12-22]. VQ is an established lossy compression technique that has been used successfully to compress signals such as speech, music, imagery, and video [21].

Vector Quantization is one of the widely used and efficient techniques for image compression. VQ has received a great attention in the field of multimedia data compression [9] since last few decades because it has simple decoding structure and can provide high compression ratio. In 1980, Linde, Buzo, and Gray proposed the VQ scheme for grayscale image compression and it has proven to be a powerful tool for both speech and digital image compression [6]. There are three major procedures in VQ, namely codebook generation, encoding procedure and decoding procedure. In the codebook generation process, various images are divided into several k-dimension training vectors. The representative codebook is generated from these training vectors by the clustering techniques. In the encoding procedure, an original image is divided into several k-dimension vectors and each vector is encoded by the index of codeword by a table look-up method. The encoded results are called an index table.

During the decoding procedure, the receiver uses the same codebook to translate the index back to its corresponding codeword for reconstructing the image. The principle involves encoding a block of pixels, called a vector, rather than encoding each pixel individually. A vector quantizer Q of dimension k and size N is defined as a mapping from vec-

ISSN:2319-7900

tors in k-dimensional Euclidean space R^k containing N output or reproduction vectors called code-words. Thus:

$$Q = R^k k \rightarrow w$$

Where W = (w_n; n = 1, 2. . . N) is the set of reproduction vectors (feature maps) called a codebook.

Vector quantization can also be seen as a combination of two functions: an encoder, which views the input vector v and generates the address of the reproduction vector specified by Q (v); and a decoder, which uses this address to generate the reproduction vector w. To the N code-words of the vector quantizer corresponds N regions of R^k (clusters, Voronoi partition), denoted Rn; n = 1, 2. . . N, The nth region is defined by:

$$R_n = \left\{ v \in \frac{R^k}{Q(v)} = w_n \right\}$$

and represents the subset of vectors of R^k which are well matched by the codeword w_n, of the codebook. At the encoding end, the address of the codeword most closely describing the vector to be coded, in terms of a predefined distance, is transmitted. A copy of the codebook is stored in a ROM in the decoder device. Then the decompressed data is generated simply by reading the codeword corresponding to the transmitted address. One can notice that the decoding stage is very simple. The bit rate is given by $R = (\log_2 N)/n$, where n is the block size and N is the codebook size. A major problem of VQ is the codebook generation [6, 9, 11, 13, 16-18].The codebook is generated using a clustering algorithm which selects the most significant vectors of a training set in order to minimize the coding error when all the training set vectors are encoded. So far, several codebook design algorithms had been proposed [1, 19] to design the VQ codebooks. Among them the LBG and Generalized Lloyd Algorithm (GLA) algorithm is the most commonly used method for VQ codebook design.

The LBG algorithm is the most cited and widely used algorithm on designing the VQ codebook. It is the starting point for most of the work on vector quantization. The performance of the LBG algorithm is extremely dependent on the selection of the initial codebook. In conventional LBG algorithm, the initial codebook is chosen at random from the training data set. It is observed that some-time it produces poor quality codebook. Due to the bad codebook initialization, it always converges to the nearest local minimum. This problem is called the local optimal problem [10]. In addition, it is observed that the time required to complete the iterations depends upon how good the initial codebook is. In literature [10-20], several initialization techniques have been reported for obtaining a better local minimum.

The LBG algorithm has two disadvantages. First the LBG algorithm only guarantees a locally optimal codebook. Second the codeword generation process needs a great deal of calculation. To reduce the computational complexity of code-words search, many fast search algorithms have been developed, such as Triangle Inequality Eliminating rule [3], Partial Distortion Search [4], Mean ordered Partial codebook Search [5] and Principal Component Analysis [6].

To make an effective initial codebook there have been several methods proposed, such as the pair-wise nearest neighbor (PNN) algorithm [7]. The obvious way in PNN algorithm is to explicitly find each point's nearest neighbor, but it is inefficient to reduce the cost order of the closest pair computation O (N log N) in training size N. The Enhanced LBG algorithm (ELBG) [8] with the same initial codebook can reduce a few numbers of iterations than LBG. But ELBG algorithm has the same computational complexity as LBG algorithm. A method of image thresholding by using cluster organization from the histogram of an image have been proposed, one of the well-known methods is Otsu'smethod [9].

By maximizing the criterion function, the variances between two classes can be separated as far as possible and the variances within both classes will be as minimal as possible [21]. Other than this various scheme include mean-distance-ordered partial codebook search (MPS), enhance LBG (ELBG), neural network based techniques, genetic-based algorithms, principal component analysis (PCA) approaches, tabu search (TS) schemes, are used to design codeword displacement methods and so on [10-21].

In order to alleviate problems associated with the codebook initialization for the LBG algorithm, in this paper, we have proposed a novel codebook initialization technique. In the initialization process, we have first chosen a highest level approximate image from the original image using image pyramid [14] and subsequently the selected highest level approximated image is decomposed into blocks to select as the initial codebook for codebook generation. The selected initial codebook is trained into an improve one through several iterative processes. The proposed algorithm has been implemented and tested on a set of standard test images and the performance is compared with respect to the standard LBG algorithm [22].

The rest of the paper is organized as follows. A brief overview VQ and LBG is given in section 2. The proposed methodology is elaborated in section 3. Experimental results are presented in section 4 to discuss the relative performance of the proposed scheme. Finally, conclusions are given in section 5.

2. LBG Algorithm Based Vector Quantization

ISSN:2319-7900

The concept of VQ is based on Shannon's rate-distortion theory where it says that the better compression is always achievable by encoding sequences of input samples rather than the input samples one by one. In VQ based image compression, initially image is decomposed into non-over lapping sub image blocks. Each sub block is then converted into one-dimension vector which is termed as training vector. From all these training vectors, a set of representative vectors are selected to represent the entire set of training vectors

vectors. As stated earlier, the VQ process is done in the following three steps namely (i) codebook design, (ii) encoding process and (iii) decoding process [23-25].

Linde et al. proposed a GLA which is also called Linde-Buzo-Gary algorithm, requires an initial codebook to start in 1980 [6]. Codebook is generated using a training set of images [26]. The set is representative of the type of images that are to be compressed. There are different methods like Random Codes and Splitting [12], in which the initial code book can be obtained. This initial codebook is obtained by the splitting method in LBG algorithm and set as the average of the entire training sequence. This code vector is then split into two. The iterative algorithm is run with these two vectors as the initial codebook. The final two code vectors are splitted into four and the process is repeated until the desired number of code vectors is obtained.

The flowchart of LBG clustering algorithm is shown in Figure 1. After codebook design process, each codeword of the codebook is assigned a unique index value. Then in the

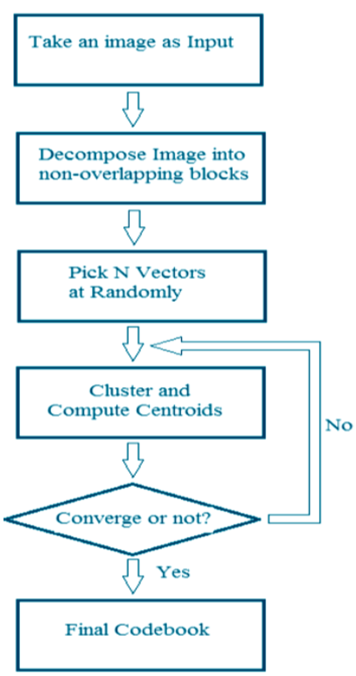

Figure 1: The flowchart of LBG clustering algorithm

encoding process, any arbitrary vector corresponding to a block from the image under consideration is replaced by the index of the most appropriate representative codeword. The matching is done based on the computation of minimum squared Euclidean distance between the input training vector and the codeword from the codebook. So after encoding process, an index table is produced. The codebook and the index-table is nothing but the compressed form of the input image. In decoding process, the codebook which is available at the receiver end too, is employed to translate the index back to its corresponding codeword. This decoding process is simple and straight forward. Figure 2 shows the schematic diagram of VQ encoding-decoding process [3, 10, 12-17, 20, 23-24].

LBG is an easy and rapid algorithm. However, it has the local optimal problem which is that for a given initial solution, it always converges to the nearest local minimum. In other words, LBG is a local optimization procedure.

Therefore, scholars proposed many approaches to solve this problem, such as directed-search binary-splitting (DSBS), mean-distance-ordered partial codebook search (MPS), double test of principal components (DTPC), enhance LBG (ELBG), centroid neural network adaptive resonance theory (CNN-ART), fast-searching algorithm using

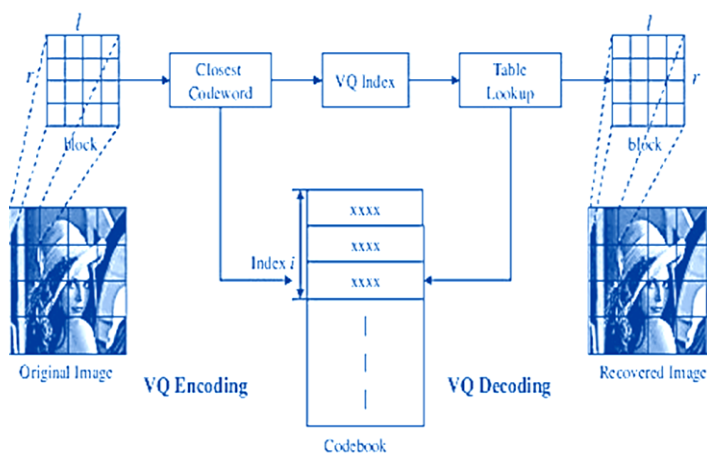

Figure 2: The Schematic diagram of VQ encoding-decoding process

projection and inequality (FAUPI), GA-based algorithm, evolution-based tabu search approach (ETSA), PNM, codebook generation algorithm using codeword displacement (CGAUCD) and so on. The discussed schemes are designed to improve the distortion of the reconstructed image and the others are designed to reduce the computation time of the training procedure [23, 26-29].

3. Methodology

Let the size of original image be $N \times N$. In VQ algorithm, each image is divided into sub-images or blocks of size $n \times n$. Therefore, there are $N_b=[N/n]\times[N/n]$ sub-images or blocks called input vectors as $X=\{x_1,x_2,\ldots, x_{Nb}\}$, where x_i is the input vector with the dimension of $n\times n$, *i.e.* $x_i=(x_{i1},x_{i2},\ldots,x_{i(n\times n)})$. In the next step, the codebook $C = \{c_1,c_2,\ldots, c_{Nc}\}$ is designed where c_i is the ith codeword as $c_i=(c_{i1},c_{i2},\ldots, c_{i(n\times n)})$ with the length of $n\times n$ and N_C is the number of code-words or size of codebook $(N_b>>N_C)$. Vector quantization method assigns each input vector to one of the code-words in the codebook based on the maximum similarity, i.e. minimum Euclidean distance [24]. The distortion function D is defined as

$$D(C) = \frac{1}{N_b}\sum_{j=1}^{N_c}\sum_{i=1}^{N_b}\mu_{ij}\left\|x_i - c_j\right\|^2 \quad (1)$$

Where μ_{ij} are called the index coefficients which should satisfy the following two conditions:

$$\sum_{j=1}^{N_c}\mu_{ij} = 1, \quad \forall i \in \{1,2\ldots, N_b\} \quad (2)$$

$$\mu_{ij} = \begin{cases} 1 & if\ x_i\ is\ the\ jth\ region\ (R_i) \\ 0 & else \end{cases} \quad (3)$$

In order to design codebook optimally, one should consider the following points:

(a) The region R_j, $j=1,2,\ldots,N_C$, which represents the input vectors belonging to the codeword c_j must satisfy

$$R_j = \{x \in X: d(x,c_j) < d(x,c_k), \quad \forall k \neq j\}$$

Where $d(x,c_j)$ is the Euclidean distance between the vectors x and c_j.

(b) The codeword c_j is the centroid of the j^{th} region (R_j) which is computed as

$$c_j = \frac{1}{N_j} i \quad for\ x_i \in R_i \quad (4)$$

Where, N_j is the number of member vectors of R_j.

One of the key points of VQ is to generate a good codebook such that the distortion between the original image and the reconstructed image is the minimum. Moreover, since the codebook generation procedure is a time consuming process, how to reduce the computation time is another important issue for the VQ codebook generation [8, 12, & 25].

The compression algorithm can be measured by certain performances such as Compression Ratio (CR), Mean square error (MSE), Peak Signal to Noise ratio (PSNR) [26-29]. They used a mapping function to partition training vectors into N clusters. The mapping function is defined as $R^k \rightarrow CB$. Let $X = (x_1, x_2 \ldots x_k)$ be a training vector and $d(X, Y)$ be the Euclidean distance between any two vectors. The iteration of GLA for a codebook generation is given as follows [4]:

Step 1: Randomly generate an initial codebook CB_0.

Step 2: $i = 0$.

Step 3: Perform the following process for each training vector.

- Compute the Euclidean distances between the training vector and the code-words in CB_i. The Euclidean distance is defined as

$$d(X,C) = \sqrt{\sum_{t=1}^{k}(x_t - c_t)^2} \quad (5)$$

- Search the nearest codeword among CB_i.

Step 4: Partition the codebook into N cells.

Step 5: Compute the centroid of each cell to obtain the new codebook CB_{i+1}.

Step 6: Compute the average distortion for CB_{i+1}. If it is changed by a small enough amount since the last iteration, the codebook may converge and the procedure stops. Otherwise, $i = i + 1$ and go to **Step 3.**

Here, we use five training vectors as an example to demonstrate how to train a codebook. The training vectors are

Table 1: Five Training Vectors

	x_1	x_2	x_3	x_4
X_1	241	192	21	156
X_2	212	76	123	36
X_3	10	220	108	233
X_4	165	108	155	41
X_5	109	52	19	247

shown in Table 1. Suppose the total number of the code-words in a codebook is three, namely $N = 3$. First, we randomly generate an initial codebook CB_0 as shown in Table 2, [4].

Next, the scheme computes the distances between the training vector and the code-words among CB_0. For example, the distance between X_1 and C_1 is $d(X_1, C_1) = \sqrt{(241-32)^2 + (192-177)^2 + ... + (156-210)^2} = 248.41$. From Table 1, we can see that the nearest code of X_1 is C_3 for $i = 0$. The training vectors which have the same nearest codeword are partitioned into the same cell. The scheme computes the centroid of each cell to obtain the new codebook. In this example, X_1 and X_4 that have the same nearest codeword C_3 are partitioned into the same cell. The centroid of the two vectors is (203, 150, 88, 98.5) that is the third codeword of the new generated codebook CB_1. The procedure is repeated until the codebook is converged. The final codebook is CB_3.

Table 2: An Example of GLA Algorithm

i	X	Nearest Codeword	Euclidean Distance C_1	C_2	C_3
0	X_1	C_3	248.41	225.942	216.959
	X_2	C_2	270.7	99.6444	126.831
	X_3	C_1	63.93	351.763	255.421
	X_4	C_3	226.17	146.952	86.9368
	X_5	C_1	195.7	243.563	263.989
1	X_1	C_3	211.99	197.74	104.896
	X_2	C_2	268.35	0	103.384
	X_3	C_1	107.4	317.134	246.25
	X_4	C_2	244.72	65.437	104.896
	X_5	C_1	107.4	257.919	212.728
2	X_1	C_3	211.99	134.142	0
	X_2	C_2	268.35	68.9227	197.74
	X_3	C_1	107.4	267.536	260.083
	X_4	C_2	244.72	79.9229	209.793
	X_5	C_1	107.4	223.534	212.859
3	X_1	C_3			
	X_2	C_2			
	X_3	C_1			
	X_4	C_2			
	X_5	C_1			

CB_i				
CB_0				
C_1	32	177	143	210
C_2	196	16	46	24
C_3	180	130	212	101
CB_1				
C_1	59.5	136	63.5	240
C_2	212	76	123	36
C_3	203	150	88	98.5
CB_2				
C_1	59.5	136	63.5	240
C_2	206	125.3	99.67	77.67
C_3	241	192	21	156
CB_3				
C_1	59.5	136	63.5	240
C_2	206	125.3	99.67	77.67
C_3	241	192	21	156

4. Results

The tested image was a picture ”Gandhi” ($N1 \times N2 = 512 \times 512$ frame of size, 256 grey levels for each pixel and blocks of image 4×4 pixels) as shown in Figure 3. The experiment shows realization of the LBG design: with nonlinear neural predictors described in this paper. In Figure 3, we show the reconstructed “Gandhi” image for the codebook size = 512, twenty loops and the FSCL algorithm. In this case MSE = 35.0054.

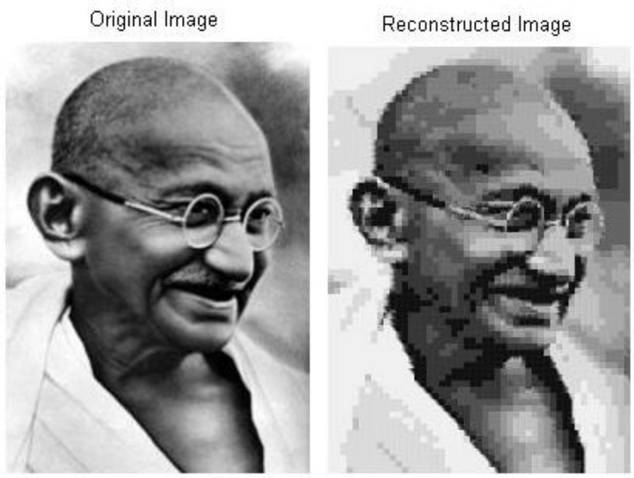

Figure 3: Original and Reconstructed "Gandhi" Image

5. Conclusions

In recent days, vector quantization (VQ) has emerged as an effective tool in the area of image compression. In earlier. days, the design of a vector quantizer was considered to be a challenging problem due to the need for multi-dimensional integration. LBG algorithm is based on training sequences. The use of training sequences bypasses the need for multi-dimensional integration

In this study, LBG-VQ has been used for the image compression applications. An image has been compressed and then reconstructed as shown above. This technique, as lossy data compression technique, has quite high value of MSE. The performance, in this study, is found to be upto the mark.

References

[1] Ong Zhang, Xiaofei Zhang, and, Shun Cao, "Analysis and evaluation of some image compression techniques," The Fourth International Conference/Exhibition on High Performance Computing in the Asia-Pacific Region, Proceedings. Vol. 2, pp. 799-803. 2000.

[2] Arup Kumar Pal and AnupSar, "An Efficient Codebook Initialization Approach For LBG Algorithm," International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.1, No.4, August 2011.

[3] Chang-Chin Huang, Du-Shiau Tsai and Gwoboa-Horng, "A Fast VQ Codebook Generation Algorithm Based on Otsu Histogram Threshold,"Prco. International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSP '08, No. 3-4, pp. 1141- 1145, 2008.

[4] Tzu-Chuen Lu and Ching-Yun Chang, "A Survey of VQ Codebook Generation," Journal of Information Hiding and Multimedia Signal Processing, Ubiquitous International, Vol.1, No.3, July 2010.

[5] Ming Yang and NikolaosBourbakis, "An Overview of Lossless Digital Image Compression Techniques," 48th Midwest Symposium on Circuits and Systems, Vol. 2, pp. 1099- 1102, 2005.

[6] Y. Linde, A.Buzo, and R.M Gray," An algorithm for vector quantizer design," IEEE on comm., Vol. 28, No.1, pp.84-95, 1980.

[7] P. Veprek and A.B. Bradley, "An improved algorithm for vector quantizer design,"Vol. 7, No.9, pp. 250-252, 2000.

[8] R.M.Gray,"vector quantization," IEEE ASSP Magazine vol. 1, no-2, pp.4-29, April 1984.

[9] Sin-Horng Chen and W.M.Hsieh, "Fast algorithm for VQ codebook design," IEE Proceedings I Communications, Speech and Vision, Vol.138, No. 5, pp. 357-362, 1991.

[10] Yu-Chen Hu, Bing-Hwang Su,andChih-Chiang Tsou, "Fast VQ codebook search algorithm for grayscale image coding," Image and Vision Computing, Vol. 26,No. 5, pp. 657-666, May 2008.

[11] N Akrout, R Prost and R Goutte, "Image compression by vector quantization: a review focused on codebook generation," Image and Vision Computing, Vol. 12, No. 10, pp.627-637, Dec. 1994.

[12] PierangelaCicconi, Emmanuel Reusens, FrédéricDufaux, IoleMoccagatta, Bruno Rouchouze, TouradjEbrahimi and Murat Kunt, "New trends in image data compression," Computerized Medical Imaging and Graphics, Vol. 18, No.2, pp.107-124, April 1994.

[13] MajidRabbani and Rajan Joshi, "An overview of the JPEG 2000 still image compression standard," Signal Processing: Image Communication, Vol.17, No.1, pp.3-48, Jan. 2002.

[14] Yu-Chen Hu and Chin-Chen Chang, "Low complexity index-compressed vector quantization for image

ISSN:2319-7900

compression," IEEE Transactions on Consumer Electronics, Vol.45, No.1, pp. 219- 224, 1999.

[15] Wenhua Li and E. Salari , "A fast vector quantization encoding method for image compression," IEEE Transactions on Circuits and Systems for Video Technology,Vol.5, No.2,pp.119-123,1995.

[16] H. Matsumoto, K. Sasazaki, and Y. Suzuki, "Color image compression with vector quantization, IEEE Conference on Soft Computing in Industrial Applications, 2008. SMCia '08, pp. 84 - 88, 2008.

[17] M. Shabanifard and M.G. Shayesteh, "A New Image Compression Method Based on LBG Algorithm in DCT Domain," 7th Iranian Int. conf. on Machine Vision and Image Processing (MVIP), pp. 1- 5, 2011.

[18] H. Ichihara, Y. Iwamoto, Yoshikawa, and T. Inoue, "Test Compression Based on Lossy Image Encoding," 2011 20th Asian Test Symposium (ATS), pp. 273- 278, 2011.

[19] S. X. Chen, F.W. Li, and W.L.Zhu, "Fast searching algorithm for vector quantization based on features of vector and subvector," IET Image Processing, Vol.2, No.6, pp. 275- 285, 2008.

[20] Chih-Tang Chang, Yi-ChingLiaw, and Jim Z.C. Lai, "A novel approach for fast codebook re-quantization," Pattern Recognition, Vol. 41, No. 9, Sep. 2008, pp.2956-2963.

[21] KannanPanchapakesan, "Image processing through vector quantization,"Ph.D. Thesis, The University of Arizona, 2000.

[22] Jianhua Lin, "Vector quantization for image compression: Algorithms and performance," Ph.D. Thesis, Brandeis University, 1992.

[23] Sharon MalkaPerlmutter, "Image compression using vector quantization: Algorithms and quality evaluation," Ph.D. Thesis, Stanford University, 1996.

[24] Karen Louise Oehler, "Image compression and classification using vector quantization," Ph.D. Thesis, Stanford University, 1993.

[25] Mei-Yin Shen and C.-C.JayKuo, "Review of Postprocessing Techniques for Compression Artifact Removal," Journal of Visual Communication and Image Representation, Elsevier, Vol. 9, No. 1, pp.2–14, Mar. 1998.

[26] Dong Liu, Xiaoyan Sun, and Feng Wu, "Inpainting with image patches for compression," Journal of Visual Communication and Image Representation, Vol.23, No.1, pp.100-113, Jan. 2012.

[27] H. Tamura and N. Nokoya, "Image database systems: A survey," Patt. Recognition, 17, pp. 29–43, 1984.

[28] AsmitaA.Bardekarand P.A.Tijare, "Implementation of LBG Algorithm for Image Compression," International Journal of Computer Trends and Technology, Vol.2, No.2, pp.86-91, 2011.

[29] LINDE Y., BUZO A., GRAY R.M.: 'An algorithm for vector quantizer design', IEEE Trans. Commun., 1980, 28, (1), pp. 84–95.

Biographies

MANOJ KUMAR, received the B.E. degree in Computer Science & Engineering from Veera College of Engineering, Bijnor from MJP University, Bareilly. Currently, he is working as faculty in Computer Science & Engineering Department in K.L.Polytechnic, Roorkee, U.K. He is pursuing M.Tech in Computer Science & Engineering from NITTTR, Chandigarh. Email: mk_saini2005@yahoo.co.in

POONAM SAINI, received the B.Tech. degree in Information Technology and M.Tech in Software Engineering and submitted Ph.D. at Kurukshetra University, Kurukshetra. Currently, she is working as Assistant Professor in Computer Science & Engineering, NITTTR Chandigarh. Her area of specialization includes Distributed Computing, Software Engineering, Network Security and Cryptography.