ERROR VECTOR ROTATION USING KEKRE TRANSFORM FOR EFFICIENT CLUSTERING IN VECTOR QUANTIZATION

H. B. Kekre[1], Tanuja K. Sarode[2] and Jagruti K. Save[3]

[1]Professor, Mukesh Patel School of Technology Management and Engineering, NMIMS University, Vileparle(w) Mumbai, India
[2]Associate Professor, Thadomal Shahani Engineering College, Bandra(W), Mumbai, India
[3]Ph.D. Scholar, MPSTME, NMIMS University, Associate Professor, Fr. C. Rodrigues College of Engineering, Bandra(W), Mumbai, India

ABSTRACT

In this paper we present an improvisation to the Kekre's error vector rotation algorithm for vector quantization (KEVR). KEVR gives less distortion as compared to well known Linde Buzo Gray (LBG) algorithm and Kekre's Proportionate Error (KPE) Algorithm. In KEVR the error vector sequence is the binary representation of numbers. Since the cluster orientation depends on the changes in the binary numbers in the sequence, the cluster orientation changes slowly. To overcome this problem, in the proposed method Kekre's transform matrix is used. It is preprocessed and then used to generate the error vector sequence. The proposed method is tested on different training images of size 256x256 for code books of sizes 128, 256, 512, 1024. Our result shows that proposed method gives less MSE (Mean Squared Error), PSNR (Peak Signal to Noise Ratio) compared to LBG, KPE and KEVR..

KEYWORDS: *Vector Quantization, Codebook, Code Vector, Data Compression, Encoder, Decoder, Clustering.*

I. INTRODUCTION

Vector Quantization (VQ) is an efficient and simple approach for data compression [1] [2][3]. Since it is simple and easy to implement, VQ has been widely used in different applications, such as pattern recognition [4], face detection [5], image segmentation [6] [7] [8], speech data compression [9], Content Based Image Retrieval (CBIR) [10], tumor detection in mammography images [11][12] etc. Vector quantization is a lossy image compression technique. There are three major procedures in VQ, namely codebook generation, encoding procedure and decoding procedure. In the codebook generation process, image is divided into several *k*-dimensional 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*-dimensional 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. One of the key points of VQ is to generate a good codebook [13] such that the distortion between the original image and the reconstructed image is the minimum. In order to find the best-matched codeword in the encoder, the ordinary VQ coding scheme employs the full search algorithm, which examines the Euclidean distance between the input vector and all codeword in the codebook. This is time consuming process. To overcome this, a fast search algorithm is reported for VQ based image compression [14] [15]. Even DCT (Discrete Cosine

Transform) based method can be used to generate the codebook [16]. It is also possible to reduce the processing time [17] [18] and computational complexity [19] for codebook generation.

1.1. Related Work

To generate codebook there are various algorithms. The most commonly used method in VQ is the Generalized Lloyd Algorithm (GLA) which is also called Linde-Buzo-Gray (LBG) algorithm [20]. However, LBG has the local optimal problem, and the utility of each codeword in the codebook is low. The codebook guarantees local minimum distortion but not global minimum distortion [21]. To overcome the local optimal problem of LBG, Giuseppe Patan´e a and Marco RussoPatane [22] proposed a clustering algorithm called enhanced LBG. Further modification to LBG method is done by using image pyramid [23]. It is also possible to reduce the computational complexity in LBG [24]. This paper aims to provide an improvement to the Kekre's error vector rotation algorithm (KEVR). We have used Kekre transform matrix to generate the error matrix. The paper also compares the proposed algorithm with LBG and Kekre's proportionate error algorithm (KPE), with respect to mean squared error (MSE) and peak signal to noise ratio (PSNR). In the next section we discuss LBG, KPE, KEVR algorithms. Section III gives some information on Kekre Transform. Proposed methodology is explained in Section IV, followed by results and conclusion in sections V and VI respectively.

II. CODEBOOK GENERATION ALGORITHMS

2.1. Linde, Buzo and Gray (LBG) Algorithm

In 1980, Linde et al. proposed a Generalized Lloyd Algorithm (GLA) which is also called Linde-Buzo-Gray (LBG) algorithm. In this algorithm, all the training vectors are clustered using minimum distortion principle. Initially all training vectors will form a single cluster. The centroid of this cluster is calculated which will become the first code vector. Constant error is added to this code vector to form two trial code vectors say v_1 and v_2. Each training vector belongs to one cluster depending upon the closeness with the trial code vector. Thus initial single cluster is divided into two clusters. The cluster centroids are calculated and they will form the set of code vectors. The process is repeated for each cluster till the code book of desired size is prepared.

2.2. Kekre's Proportionate Error Algorithm (KPE) [25]

Here instead of constant error as in LBG algorithm, proportionate error is added to the code vector. It is seen that in LBG algorithm the cluster formation is elongated about 135^0. So the clustering is inefficient. In KPE, the magnitude of the coordinates of centroid decides the error ratio. While adding proportionate error a safe guard is also introduced so that two trial code vectors do not go beyond the training vector space. This method gives better results compared to LBG method.

2.3. Kekre's Error Vector Rotation Algorithm (KEVR) [26]

In this algorithm, from the initial codevector the two trial code vectors say v1 & v2 are generated by adding and subtracting error vector rotated in k-dimensional space at different angle. Then two clusters are formed on the basis of closeness of the training vectors with trial code vectors. The centroids of two clusters formed the codevectors of the code book. This modus operandi is repeated for every cluster and every time to split the clusters error Ei is added and subtracted from the codevector. Error vector Ei is the ith row of the error matrix of dimension k. The error vectors matrix E is given in Equation 1. The error vector sequence have been obtained by taking binary representation of numbers starting from 0 to k-1 and replacing 0's by 1's and 1's by -1's.

$$E = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ . \\ . \\ e_k \end{bmatrix} = \begin{bmatrix} 1\ 1\ 1\ 1 \ldots\ldots\ldots\ 1\ 1 & 1 & 1 \\ 1\ 1\ 1\ 1 \ldots\ldots\ldots\ 1\ 1 & 1 & -1 \\ 1\ 1\ 1\ 1 \ldots\ldots\ldots\ 1\ 1 & -1 & 1 \\ 1\ 1\ 1\ 1 \ldots\ldots\ldots\ 1\ 1 & -1 & -1 \\ \ldots\ldots\ldots\ \ldots\ldots\ldots\ \ldots\ldots\ldots & .. & \\ . & & \\ \ldots\ldots\ldots\ \ldots\ldots\ldots\ \ldots\ldots\ldots & .. & \end{bmatrix} \quad (1)$$

III. KEKRE TRANSFORM MATRIX[27]

Kekre transform matrix can be of any size NxN, which need not have to be in powers of 2(as is the case with most of other transforms). All upper diagonal and diagonal values are one, while the lower diagonal part except the values just below diagonal is zero. Generalized NxN Kekre transform matrix can be given as shown in equation 2.

$$K(NXN) = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & 1 \\ -N+1 & 1 & 1 & \cdots & 1 & 1 \\ 0 & -N+2 & 1 & \cdots & 1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 1 \\ 0 & 0 & 0 & \cdots & -N+(N-1) & 1 \end{bmatrix} \quad (2)$$

The formula for generating the term Kxy of Kekre transform matrix is given in equation 3.

$$Kxy = \begin{cases} 1 & x \le y \\ -N+(x-1) & x = y+1 \\ 0 & x > y+1 \end{cases} \quad (3)$$

Kekre transform matrix is orthogonal, asymmetric and non involutional.

IV. PROPOSED METHOD

In this method Kekre transform matrix is used for generation of error vectors. But before using this matrix some preprocessing is done on it. The matrix is flipped horizontally and vertically as explained in algorithm given below.

4.1. Algorithm to flip the matrix

Let A= Input matrix

Let K= Output matrix

For i= 1 to N(no. of rows)

For j=1 to N(no. of columns)

K(i,j) = A(N+1-i,N+1-j)

End

End

After applying the above algorithm we get the transform matrix as given in equation 4.

$$K(NXN) = \begin{bmatrix} 1 & -N+(N-1) & \cdots & 0 & 0 & 0 \\ 1 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & \cdots & 1 & -N+2 & 0 \\ 1 & 1 & \cdots & 1 & 1 & -N+1 \\ 1 & 1 & \cdots & 1 & 1 & 1 \end{bmatrix} \quad (4)$$

Now we will keep only signs in this matrix so we get the matrix as shown below in equation 5.

$$K(\mathrm{NxN}) = \begin{bmatrix} 1 & -1 & \cdots & 0 & 0 & 0 \\ 1 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & \cdots & 1 & -1 & 0 \\ 1 & 1 & \cdots & 1 & 1 & -1 \\ 1 & 1 & \cdots & 1 & 1 & 1 \end{bmatrix} \quad (5)$$

The rows of the above matrix are used to generate the error vectors.

4.2. Proposed Algorithm – Kekre's Error Vector Rotation Algorithm using Kekre Transform (KEVRK)

1. Divide the image into non overlapping blocks of size 2 x 2 and convert each block to training vector of size 12 x 1 forming the training vector set.
2. Initialize i=1;
3. Calculate the centroid (first codevector) of training vectors.
4. Generate two trial code vectors $V_1$and V_2 by adding and subtracting error vector Ei (i^{th} row from the matrix given in equation 5) from the each code vector.
5. Calculate Euclidean distance between each training vector and code vector. On the basis of this distance, each cluster is split into two clusters.
6. Calculate the centroid (codevector) for each cluster.
7. Increment i by one and repeat step 4 to step 6 for each codevector.

Repeat above procedure till the codebook of desire size is obtained.

V. RESULTS

The algorithms discussed above are implemented using MATLAB 7.0 on Pentium IV, 1.66GHz, 1GB RAM. To test the performance of these algorithms nine color images as shown in Figure 1 belonging to different classes are used. The images used belong to class Animal, Bird, Vehicle, Flowers, and Scenery etc.

Figure 1. Testing Images

To implement proposed algorithm, Kekre transform matrix of dimension 12 is generated as given in equation 5. Table 1 shows the comparison of LBG, KPE, KEVR and proposed algorithm (KEVRK) for codebook size 128 and 256 with respect to MSE, PSNR for the training images. Table 2 shows the comparison of LBG, KPE, KEVR and KEVRK for codebook size 512 and 1024 with respect to MSE, PSNR for the training images. Figure 2 shows the results of LBG, KPE, KEVR, and KEVRK for code book size of 256 on bird image. Figure 3 shows the average MSE performance for LBG, KPE, KEVR and proposed technique KEVRK for different code book (CB) sizes.

Table 1. Comparison of LBG, KPE, KEVR and Proposed Algorithm for Codebook size 128 and 256 with respect to MSE, PSNR for the testing images.

Images	Param-eters	LBG	KPE	KEVR	KEVRK	LBG	KPE	KEVR	KEVRK
		128				256			
LENA	**MSE**	190.26	170.72	94.48	**88.81**	173.46	134.12	72.77	**71.48**
	PSNR	25.34	25.81	28.38	**28.65**	25.74	26.86	29.51	**29.60**
Airplane	**MSE**	221.63	189.22	127.96	**112.81**	201.96	139.96	100.81	**77.96**
	PSNR	24.67	25.36	27.06	**27.61**	25.08	26.67	28.10	**29.21**
Bus	**MSE**	652.57	521.07	338.14	**266.97**	584.22	319.99	266.43	**180.19**
	PSNR	19.98	20.96	22.84	**23.87**	20.47	23.08	23.87	**25.57**
Tiger	**MSE**	658.10	579.90	383.37	**377.25**	605.89	425.52	325.49	**279.79**
	PSNR	19.95	20.50	22.29	**22.36**	20.31	21.84	23.01	**23.66**
Bird	**MSE**	497.08	338.80	231.07	**198.00**	444.97	258.67	190.18	**155.87**
	PSNR	21.17	22.83	24.49	**25.16**	21.65	24.00	25.34	**26.20**
Ganesh	**MSE**	541.72	498.16	356.35	**349.44**	494.12	396.08	313.50	**271.48**
	PSNR	20.79	21.16	22.61	**22.70**	21.19	22.15	23.17	**23.79**
Garden	**MSE**	510.22	479.54	311.48	**300.35**	470.96	363.55	271.56	**238.20**
	PSNR	21.05	21.32	23.20	**23.35**	21.40	22.53	23.79	**24.36**
flowers	**MSE**	394.24	225.50	181.27	**177.00**	361.22	189.28	144.83	**137.62**
	PSNR	22.17	24.60	25.55	**25.65**	22.55	25.36	26.52	**26.74**
Sunset	**MSE**	520.09	313.14	204.43	**194.99**	463.60	234.18	166.57	**152.53**
	PSNR	20.97	23.17	25.03	**25.23**	21.47	24.44	25.91	**26.30**
Average	**MSE**	465.10	368.45	247.62	**229.51**	422.27	273.48	205.79	**173.90**
	PSNR	21.79	22.86	24.61	**24.95**	22.21	24.10	25.47	**26.16**

Table 2. Comparison of LBG, KPE, KEVR and Proposed Algorithm for Codebook size 512 and 1024 with respect to MSE, PSNR for the testing images.

Images	Param-eters	LBG	KPE	KEVR	KEVRK	LBG	KPE	KEVR	KEVRK
		512				1024			
LENA	MSE	151.43	94.25	55.17	**54.35**	114.16	65.05	42.55	**41.37**
	PSNR	26.33	28.39	30.71	**30.78**	27.56	30.00	31.84	**31.96**
Airplane	MSE	171.74	94.29	68.34	**57.49**	131.90	58.68	48.82	**40.75**
	PSNR	25.78	28.39	29.78	**30.53**	26.93	30.45	31.24	**32.03**
Bus	MSE	496.00	191.49	141.91	**130.23**	380.52	122.59	105.92	**95.35**
	PSNR	21.18	25.31	26.61	**26.98**	22.33	27.25	27.88	**28.34**
Tiger	MSE	535.70	288.07	228.15	**206.27**	424.77	194.96	166.40	**148.95**
	PSNR	20.84	23.54	24.55	**24.99**	21.85	25.23	25.92	**26.40**
Bird	MSE	373.50	199.64	139.27	**114.52**	297.20	137.63	104.60	**85.11**
	PSNR	22.41	25.13	26.69	**27.54**	23.40	26.74	27.94	**28.83**
Ganesh	MSE	437.85	270.66	219.43	**206.08**	350.76	187.43	170.69	**150.30**
	PSNR	21.72	23.81	24.72	**24.99**	22.68	25.40	25.81	**26.36**
Garden	MSE	416.77	256.45	203.03	**187.40**	339.70	182.29	158.86	**141.73**
	PSNR	21.93	24.04	25.06	**25.40**	22.82	25.52	26.12	**26.62**
flowers	MSE	299.28	146.57	108.26	**99.12**	223.74	104.35	81.07	**71.81**
	PSNR	23.37	26.47	27.79	**28.17**	24.63	27.95	29.04	**29.57**
Sunset	MSE	386.49	180.20	125.04	**111.05**	269.51	180.20	95.66	**82.45**
	PSNR	22.26	25.57	27.16	**27.68**	23.83	25.57	28.32	**28.97**
Average	MSE	363.20	191.29	143.18	**129.61**	281.36	137.02	108.29	**95.31**
	PSNR	22.87	25.63	27.01	**27.45**	24.00	27.12	28.23	**28.79**

Remark : It is observed that KEVRK Outperforms LBG and KPE by large margin and KEVR by smaller margin for MSE and PSNR

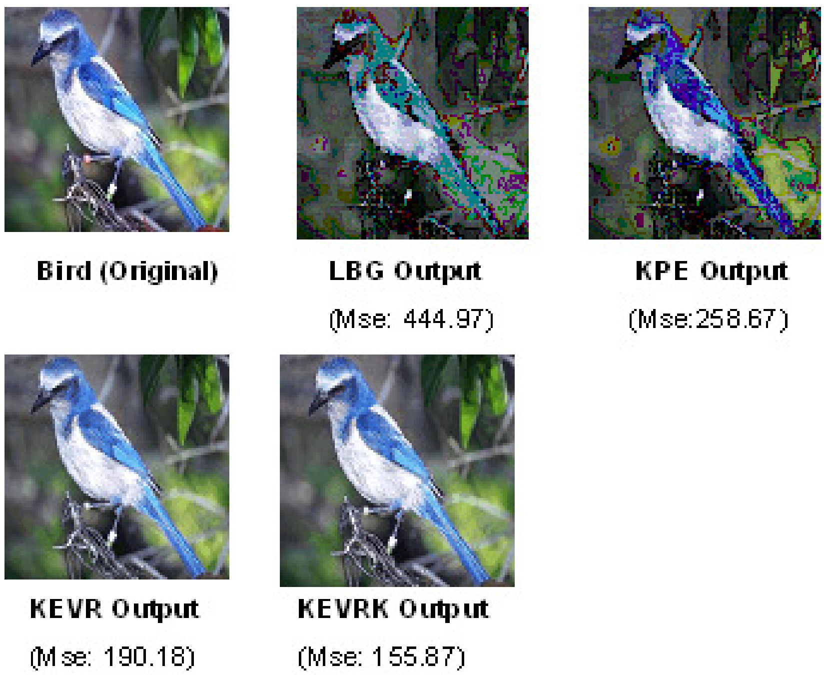

Figure 2. Results of LBG, KPE, KEVR and KEVRK from codebook size 256 on Bird image.

Remark: It may be noted that MSE reduces in order from LBG, KPE, KEVR to KEVRK

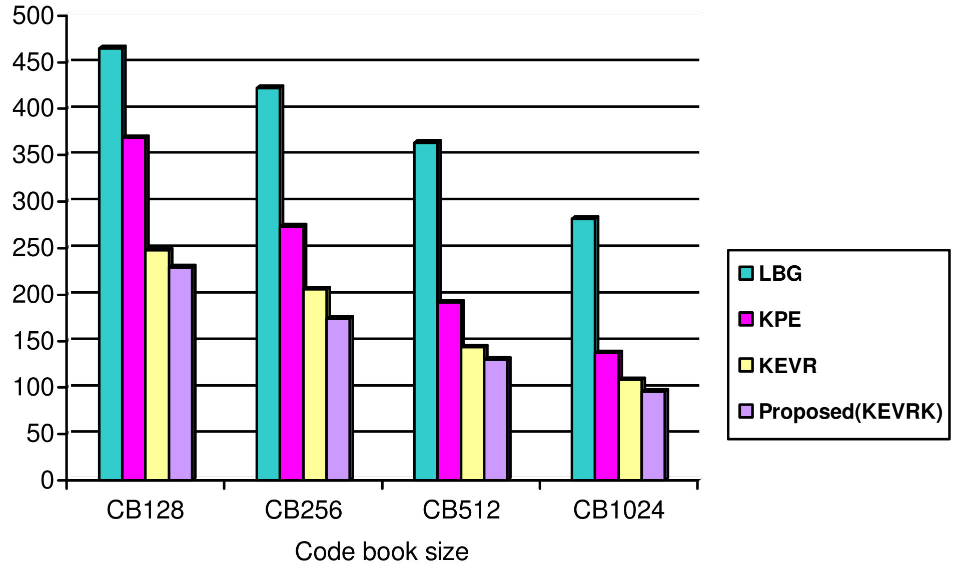

Figure 3. Average MSE performance for LBG, KPE, KEVR and Proposed (KEVRK) for different code book (CB) sizes

Remark: For proposed method average MSE is minimum.

4.1. Discussion

In LBG constant error is added with a constant angle of 45^0 to generate the code vectors so the clustering is inefficient. In KPE the error vector direction varies from 0^0 to 90^0, so that it gives an improvement over LBG. In KEVR the error vector is rotated that covers all directions giving better performance compared to LBG and KPE. To improve it further, we present the fast rotation of error vector by using Kekre transform matrix. It can be seen from the tables and figures that the proposed technique KEVRK outperforms the other methods. It can also be seen that, as the codebook size increases from 128 to 1024, the MSE decreases for all the methods.

VI. CONCLUSIONS AND FUTURE WORK

This paper aims to present an improvement to KEVR algorithm. In KEVR, Error vector matrix is the sequence of binary numbers. Since the bit change in the sequence is slow, it results in slowly changing cluster orientation. After preprocessing done on the Kekre Transform matrix, it contains values 1,-1, and 0. The row of this matrix is used as the error vector. In every two consecutive error vectors there is a two bit change. So there is a fast change in cluster orientation. This gives effective clustering. It is observed that proposed new algorithm KEVRK improves the performance of KEVR. The proposed method reduces MSE by 51% to 63% for codebook size 128 to 1024 with respect to LBG, by 30% to 40% with respect to KPE and by 8% to 16% with respect to KEVR. In future work we will investigate the performance of different similarity measures other than Euclidean distance criteria. We will further explore the different alternatives of rotation of error vector to generate the code vector.

REFERENCES

[1]. A. Gersho, R. M. Gray, "Vector Quantization and Signal Compression," Kluwer Academic Publishers, Boston, MA, 1991
[2]. R. M. Gray, "Vector quantization," IEEE ASSP Mag, Apr.1984
[3]. J. Pan, Z. Lu, and S.He Sun, "An Efficient Encoding Algorithm for Vector Quantization Based on Subvector Technique", IEEE Transactions on image processing, vol 12 No. 3 March 2003.

[4]. A. A. Abdelwahab and N. S. Muharram, "A Fast Codebook Design Algorithm Based on a Fuzzy Clustering Methodology," International Journal of Image and Graphics, vol. 7, No. 2, pp. 291-302, 2007.
[5]. C. Garcia and G. Tziritas, "Face Detection using Quantized Skin Color Regions Merging and Wavelet Packet Analysis," IEEE Trans. Multimedia, vol. 1, No. 3, pp. 264–277, Sep. 1999
[6]. H. B. Kekre, T. K. Sarode and B. Raul, "Color Image Segmentation using Kekre's Algorithm for Vector Quantization," International Journal of Computer Science (IJCS), vol. 3,No. 4, pp. 287-292, Fall 2008.
[7]. H. B. Kekre, T. K. Sarode and B. Raul, "Color Image Segmentation using Vector Quantization Techniques Based on Energy Ordering Concept," International Journal of Computing Science and Communication Technologies (IJCSCT), vol. 1, Issue 2, January 2009.
[8]. H. B. Kekre, V. A. Bharadi, S. Ghosalkar and R. A. Vora, "Blood Vessel Structure Segmentation from Retinal Scan Image using Kekre's Fast Codebook Generation Algorithm," Proceedings of the International Conference & Workshop on Emerging Trends in Technology, ACM, ICWET'2011, NY,USA, pp. 10-14, 2011.
[9]. H. B. Kekre and T. K. Sarode, "Speech Data Compression using Vector Quantization," WASET, International Journal of Computer and Information Science and Engineering, (IJECSE), vol. 2, Number 4, pp. 251-254, 2008.
[10]. H. B. Kekre, T. K. Sarode and S. D. Thepade, "Image Retrieval using Color-Texture Features from DCT on VQ Codevectors obtained by Kekre's Fast Codebook Generation," ICGST-International Journal on Graphics, Vision and Image Processing (GVIP), vol. 9, Issue 5, pp. 1-8, September 2009.
[11]. H. B. Kekre, T. K. Sarode and S. Gharge, "Detection and Demarcation of Tumor using Vector Quantization in MRI images," International Journal of Engineering Science and Technology, vol.1, Number 2, pp. 59-66, 2009.
[12]. H. B. Kekre, T. K. Sarode and S. Gharge, "Kekre's Fast Codebook Generation algorithm for tumor detection in mammography images," Proceedings of the International Conference and Workshop on Emerging Trends in Technology, ICWET'10, ACM, pp. 743-749, NY, USA, 2010.
[13]. C. M. Huang and R. W. Harris, "A Comparison of Several Vector Quantization Codebook Generation Approaches," IEEE Trans. On Image Processing, vol. 2, No. 1, pp. 108-112, 1993
[14]. Hsieh.C.H. , Lu.P.C. and Chung. J.C, "Fast Codebook Generation Algorithm for Vector Quantization of Images", Pattern Recognition Letter, 12, pp. 605-609. 1991.
[15]. H.Q., Cao, "A fast search algorithm for vector quantization using a directed graph ", IEEE Transactions on Circuits and Systems for Video Technology, vol.10, No. 4, pp. 585-593, Jun 2000
[16]. Hsieh. C.H., "DCT based Codebook Design for Vector Quantization of Images", IEEE Transactions on Circuits and Systems, Vol. 2, pp. 401-409, 1992.
[17]. H. B. Kekre and T. K. Sarode, "Centroid Based Fast Search Algorithm for Vector Quantization," International Journal of Imaging (IJI), vol. 1, No. 08, pp. 73-83, Autumn 2008
[18]. C. T. Chang, J. Z. C. Lai and M. D. Jeng, "Codebook Generation Using Partition and Agglomerative Clustering," International Journal on Advances in Electrical and Computer Engineering , Vol. 11, Issue. 3, 2011.
[19]. Chang-Chin Huang , Du-Shiau Tsai and Gwoboa Horng , "Efficient Vector Quantization Codebook Generation Based on Histogram Thresholding Algorithm ," International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSP '08, pp. 1141 - 1145
[20]. Y. Linde, A. Buzo, and R. M. Gray.: 'An Algorithm for Vector Quantizer Design," IEEE Trans. Commun., vol. COM-28, No. 1, pp. 84-95, 1980.
[21]. C. M. Huang and R. W. Harris, "A Comparison of Several Vector Quantization Codebook Generation Approaches," IEEE Trans. Image Processing, vol. 2, no. 1, pp. 108-112, 1993.
[22]. Giuseppe Patan´e a and Marco Russo, "The Enhanced LBG Algorithm," Journal of Neural Networks, vol. 14 Issue 9, November 2001 Elsevier Science Ltd. Oxford, UK
[23]. A. K. Pal and A. Sar, "An Efficient Codebook Initialization Approach for LBG Algorithm," International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.1, No.4, August 2011
[24]. Pan Z. B. , Yu G. H. , Li Y., "Improved fast LBG training algorithm in Hadamard domain," Electronics Letters, Vol. 47, Issue. 8, April 14, 2011
[25]. H. B. Kekre and T. K. Sarode, "Clustering Algorithm for codebook Generation using Vector Quantization", National Conference on Image Processing, TSEC, India, Feb 2005
[26]. H. B. Kekre and T. K. Sarode, "New Clustering Algorithm for Vector Quantization using Rotation of Error Vector," International Journal of Computer Science and Information Security,(IJCSIS), vol. 7, No. 3, pp. 159-165, 2010.

[27]. H. B. Kekre and S. D. Thepade, "Image Retrieval using Non-Involutional Orthogonal Kekre's Transform", International Journal of Multidisciplinary Research and Advances in Engg. (IJMRAE), Vol.1, No.1, pp. 189-203, Nov 2009.

Authors

H. B. Kekre has received B.E. (Hons.) in Telecomm. Engineering. from Jabalpur University in 1958, M.Tech (Industrial Electronics) from IIT Bombay in 1960, M.S.Engg. (Electrical Engg.) from University of Ottawa in 1965 and Ph.D. (System Identification) from IIT Bombay in 1970 He has worked as Faculty of Electrical Engineering and then HOD Computer Science and Engg. at IIT Bombay. For 13 years he was working as a professor and head in the Department of Computer Engg. at Thadomal Shahani Engineering. College, Mumbai. Now he is Senior Professor at MPSTME, SVKM's NMIMS University. He has guided 17 Ph.Ds, more than 100 M.E./M.Tech and several B.E./ B.Tech projects. His areas of interest are Digital Signal processing, Image Processing and Computer Networking. He has more than 450 papers in National /International Conferences and Journals to his credit. He was Senior Member of IEEE. Presently He is Fellow of IETE and Life Member of ISTE. Six Research scholars working under his guidance have been awarded Ph.D by NMIMS University. Recently fifteen students working under his guidance have received best paper awards. Currently eight research scholars are pursuing Ph.D. program under his guidance.

Tanuja K. Sarode has Received Bsc. (Mathematics) from Mumbai University in 1996, Bsc.Tech.(Computer Technology) from Mumbai University in 1999, M.E. (Computer Engineering) from Mumbai University in 2004, currently Pursuing Ph.D. from Mukesh Patel School of Technology, Management and Engineering, SVKM's NMIMS University, Vile-Parle (W), Mumbai, INDIA. She has more than 10 years of experience in teaching. She is currently working as Associate Professor in Dept. of Computer Engineering at Thadomal Shahani Engineering College, Mumbai. She is life member of IETE, member of International Association of Engineers (IAENG) and International Association of Computer Science and Information Technology (IACSIT), Singapore. Her areas of interest are Image Processing, Signal Processing and Computer Graphics. She has 100 papers in National /International Conferences/journal to her credit.

Jagruti K. Save has Received B.E. (Computer Engg.) from Mumbai University in 1996, M.E. (Computer Engineering) from Mumbai University in 2004, currently Pursuing Ph.D. from Mukesh Patel School of Technology, Management and Engineering, SVKM's NMIMS University, Vile-Parle (W), Mumbai, INDIA. She has more than 10 years of experience in teaching. Currently working as Associate Professor in Dept. of Computer Engineering at Fr. Conceicao Rodrigues College of Engg., Bandra, Mumbai. Her areas of interest are Image Processing, Neural Networks, Fuzzy systems, Data base management and Computer Vision. She has 6 papers in National /International Conferences to her credit.