Implementation of Direct Indexing and 2-V Golomb Coding of Lattice Vectors for Image Compression

R. R. Khandelwal, P. K. Purohit

Abstract: Indexing of code vectors is a most difficult task in lattice vector quantization. In this work we focus on the problem of efficient indexing and coding of indexes. Index assignment to the quantized lattice vectors is computed by direct indexing method, through which a vector can be represented by a scalar quantity which represents the index of that vector. This eliminates the need of calculating the prefix i.e. index of the radius (R) or norm and suffix i.e. the index of the position of vector on the shell of radius R, also eliminates index assignment to the suffix based on lattice point enumeration or leader's indexing . Two value golomb coding is used to enumerate indices of quantized lattice vectors. We use analytical means to emphasize the dominance of two value golomb code over one value golomb code. This method is applied to achieve image compression. Indexes of particular subband of test images like barbara, peppers and boat are coded using 2-value golomb coding (2-V GC) and compression ratio is calculated. We demonstrate the effectiveness of the 2-V GC while the input is scanned columnwise as compare to rowwise. Experimentally we also show that good compression ratio is achieved when only higher order bits of the indexes are encoded instead of complete bits.

Index Terms: Golomb coding, two value golomb coding, Lattice Vector Quantization (LVQ), indexing, image compression

I. INTRODUCTION

An image can be compressed using appropriate quantization technique on transformed coefficients followed by coding of indexes assigned to the quantized vectors. During the last few years quantization using lattice structure has attracted interest of the compression community due to its reduced memory requirement for storage and low complexity encoding and decoding algorithms. It is possible due to the regular pattern of lattice structure. But the lattice code vector's indexing pretense a main trouble. Various methods of indexing are proposed by researchers depending upon the type of lattice structure and shape of truncation. Indexing of lattice codevectors for voronoi codes is explained in [1]. When a product code is applied to assign an index to the lattice vector, the construction of index is the concatenation of two indices: one is called; prefix another is called suffix. Prefix is the index of the radius or norm and suffix is the position of vector on the shell of constant radius or norm. The complete index formed for a lattice vector by multiplexing the codes of the prefix and the suffix.

Revised Manuscript Received on July 05, 2019.

R. R. KHANDELWAL, Associate Professor Department of Electronics Engineering, Shri Ramdeobaba College of Engineering and Management, Katol Road, Nagpur- 440013, Maharashtra, India.

P. K. PUROHIT, Professor, Department of Science, National Institute of Technical Teachers' Training & Research, Bhopal, India

The indexing and encoding of prefix part, that represents the energy or norm of the vector can be easily attained. Indexing of suffix is usually done according to two different techniques. The method proposed in [2] and [3] are based on leader's indexing. Elementary vectors of hypersurface called leaders are the vectors that lead to all the other vectors lying on the hypersurface which are generated by permutations and sign changes. In order to attribute an index to a leader especially for high dimension lattice structure and norms, it is necessary to generate all leaders, which remain a highly complicated task. A method for indexing a leader based on partition function is described in [4], partition function directly gives the index of the leader and there is no need to generate all the leaders. Another approach is based on indexing vectors considering the total number of vectors lying on a given hyper surface [5][6]. A method of indexing based on product code is reported in [7] to pull out diverse information entities from the lattice vector to be indexed. Entropy coded lattice vector quantization is applied to Dead zone lattice vector quantizer codebook [8]. Codebook constitution for spherical vector quantization based on shells of the Gosset lattice and a novel approach for the code vector to index mapping are used in [9]. Direct scalar index values are assigned to the quantized vectors in [10], advantage of this method is no need to find out the address of the leaders and track the address of other vectors. Various variable length coding techniques are used to encode index of vectors such as Huffman coding [11], arithmetic coding discussed in [12][13].Theoretically entropy is a known measure to provide minimum number of bits required to present an information and the difference between the average bit rate and the entropy represents the amount of the redundancy in the code [12] . Due to shortcomings of entropy codes, redundancies always exist in practical compression systems. This is the basis of further research that has been conducted in this field. In this paper, however we focus on an efficient entropy coding technique known as Golomb coding [14] used for sequences with significant runs of zeros. In many standards Golomb–Rice code is used because of its simplicity in implementation than Huffman code [15–18]. Huffman code becomes impractical for the large number of quantization levels, because of high implementation complexity. In [19] different lengths runlength integer symbol assigned a unique binary representation, the scheme based on Golomb Coding used to represent test data compression for a system–on-chip design. A bitplane Golomb coding technique developed for Laplacian sources [20] is compare with the Golomb code developed for non negative integer sources

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication

with geometrical distribution [14]. Quantization and compression of data jointly is achieved by sharing circuits between an integrating A/D and a Golomb–Rice encoder [21], this yield a highly condense converter able to compress data. The limitation is due to counting process, converter with linearity requires a longer conversion time at each conversion and the serial read out of the final codeword. Two coding techniques Extended Golomb Coding and Modified Extended Golomb Coding (MEGC) are developed for compression of non negative integers [22]. MEGC works well for small as well as large integers unlike the Golomb Coding. The method elucidated in [23] provides reduced length of the key stream with high security level through the same coding efficiency while preserving other features of regular Ex-Golomb coding. Index compression is one of the factor on which the efficiency of Information Retrieval System (IRS) such as search engine depends, Extended Golomb Code is used to reduce the size of inverted index [24]. To improve the performances of the Golomb family of codes joint-probability-based adaptive Golomb coding (JPBAGC) scheme is reported in [25]. This scheme could be assimilated into any image or video coding system such as JPEG and the H.264-intra JPEG-based image coding system. To code the transform coefficients, Exponential Golomb coding & context adaptive variable length encoding are used [26]. A hybrid quantizer composed of a uniform scalar quantizer and a non-uniform optimal companding scalar quantizer [27], is designed for a Gaussian source and its performance is compared with uniform quantizer. For reliable and bandwidth efficient communication of multimedia information, generalized form of the family of Unary Error Correction (UEC) and Elias Gamma Error Correction (EGEC) codes to the class of rice and exponential Golomb error correction codes are developed [28] for both source coding and channel coding. In this paper, indexing method is developed to assign an integer value to a codevector, which depends upon the base value and simple calculation is used to get the base value. Finally, Golomb coding and two value Golomb coding are employed to encode the binary values of the indexes of the quantized vectors. The paper is structured as follows. Section II describes the steps for construction of lattice vector quantizer in which we elucidate the technique from the literature for each step. Section III discusses the indexing methods of codevectors along with the example of the developed method to get the index of quantized vectors. Section IV and section V explains Golomb coding and two value Golomb coding methods respectively for coding of index values. Experimental results are part of section VI which is followed by conclusion in section VII.

II. LATTICE VECTOR QUANTIZER CONFIGURATION

Global image compression system consists of mapping, quantization and symbol coding. Mapping using transform is well known redundancies reduction method in image coding. A variety of techniques can be used to quantize the transform coefficients. Configuration of lattice vector quantizer in an image coder consists of following processes:

A. Choice of optimum lattice

The first issue to be deal with in the design of lattice quantizer is the selection of best/optimum lattice. There is strong analogy between the optimum lattice selection and the sphere packing theory. This theory gives the best arrangement of equal and non overlapping spheres in a given volume in n-dimensional space. The best packing lattice structure is the one that provide the densest packing of the like spheres together. In different dimensions the best packing lattices are reported in [29].This work is based on D_8 lattice. The checkerboard lattice is defined for arbitrary dimensions as

$D_n = \{x=[x_0 \ldots\ldots x_{n-1}] \in Z^n , \sum_{i=0}^{n-1} x_i) \bmod 2=0 \}$ …..(1)

B. Selection of truncated lattice structure

The next issue that needs to give attention, after the selection of suitable lattice is the selection of finite lattice points out of infinite lattice points. Truncation of lattice points is based on isonorms. These norms may be l_1, l_2 and maximum norm defined respectively by $l_1 = \sum_{i=1}^{n}|x_i|$, $l_2 = \sum_{i=1}^{n}|x_i|$ andlattice points that have same norm value, related to truncation shape pyramidal (hyper-pyramidal for high dimensions), circular (spherical/ hyper-spherical for higher dimensions) and rectangular respectively. Fig.1 shows Pyramidal truncation of Z^2 lattice, Fig. 2 shows spherical truncation of Z^2 lattice and Fig.3 shows rectangular truncation of Z^2 lattice.

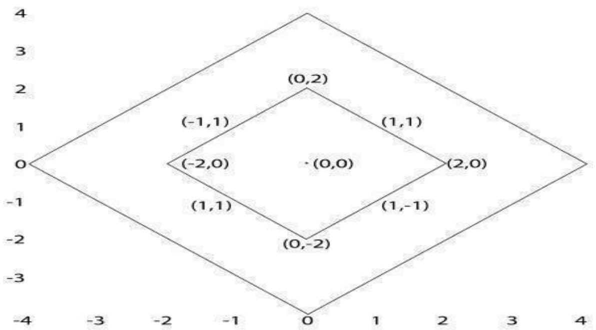

Fig 1 Pyramidal truncation of Z^2 lattice

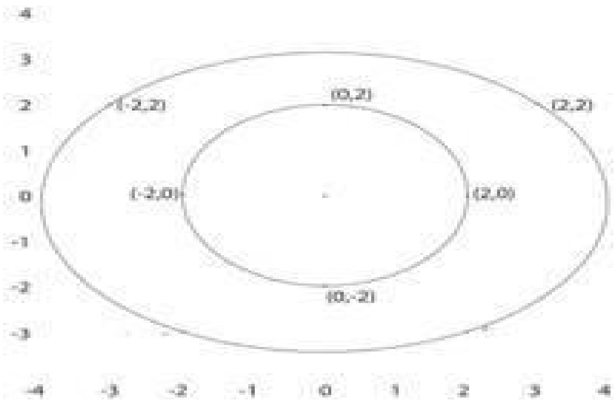

Fig 2 Spherical truncation of Z^2 lattice

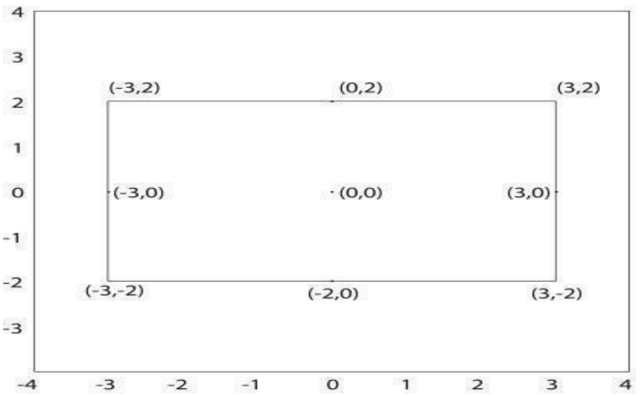

Fig 3 Rectangular truncation of Z^2 lattice

C. Number of lattice points on iso norms

Next step after truncation of lattice codebook is to determine the number of points on the truncated surface. In the literature a number of methods for the enumeration of lattice points of pyramidal shape truncation or spherical shape truncation exist. In [30] theta series is used to calculate number of points on each spherical shell and Nu series to calculate points on pyramidal shells. The theta series approach to count the lattice points on spherical shells (l2 norm) generalized to pyramidal shells (l1 norm) is explained in [31].

D. Finding closest lattice points

Algorithms to find out closest lattice points to the input vector x of the lattices D_8, E_8 and Z^4 are developed by Conway and Sloane [32]. To get closest lattice point, steps shown in the flow chart in Fig. 4 are followed.

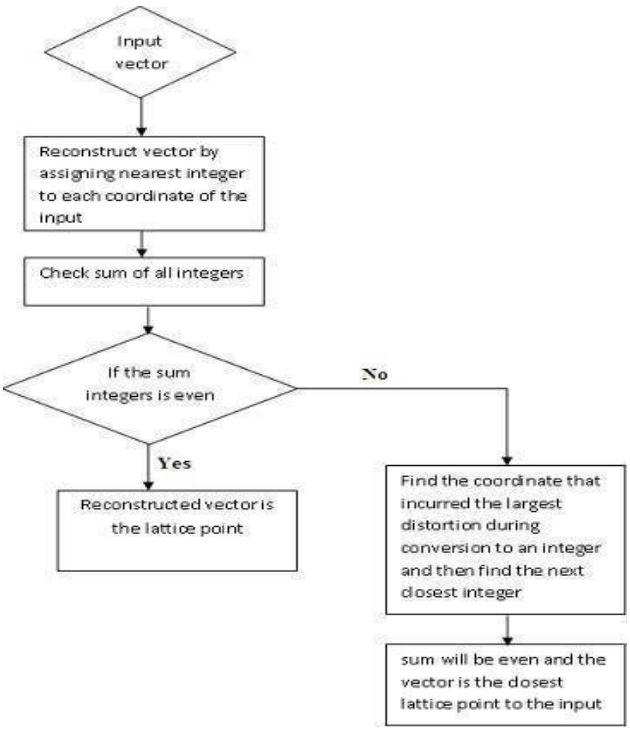

Fig. 4 Flow chart to get closest lattice point

Following example elucidate the process of calculating the closest lattice point of vectors.

Example: If two input vector points are x1= (0.2, 1.8,-0.7, 0.4, 0.3, 1.9, -1.7, 0.2) and x2 = (0.4, 0.7,-1.1, 1.8, 0.3, 0.2, -1.8, 1.9) respectively, then as per the algorithm reconstructed vectors are f(x1) = (0, 2, -1,0, 0, 2, -2, 0) and f(x2) = (0, 1, -1, 2, 0, 0, -2, 2) and their sums are, sum{f(x1)} = 1 (odd value) and sum{f(x2)} = 2 (even value), for x2 sum{f(x2)}is an even value, so the closest lattice point as per the algorithm will be f(x2) i.e. (0,1,-1, 2, 0 ,0, -2, 2), while for x1 Sum{f(x1)}is an odd value so chose the next closest integer to the coordinate that incurred the largest distortion during conversion to an integer. Here it is 0.4 , instead of 0 it should be rounded to 1. The new reconstructed vector is h(x1) = (0, 2,-1,1, 0, 2,-2, 0) with its sum, sum{h(x1)}= 2 (even value), so the closest lattice point for x1 will be (0, 2,-1,1,0, 2, -2, 0).

E. Index assignment to lattice quantized vectors

The main problem in lattice vector quantization is indexing of vectors. In literature many indexing methods are proposed for different types of lattices and their truncation. In broad sense, classification of indexing methods of lattice vectors is: Indexing based on lattice point enumeration shown in Fig. 5 and indexing based on leader's indexing shown in Fig.6. Method discussed in this paper is based on independent indexing of vectors.

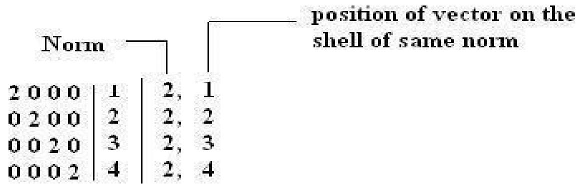

Fig 5 Indexing based on lattice points enumeration

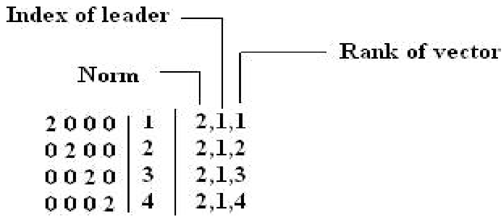

Fig 6 Indexing based on leader's addressing

F. Coding of Indices

Last stage for the design of lattice vector quantizer is coding of vector indices. Lossless coding techniques are used to code the indices. Very popular choice for lossless coding is Huffman coding. It is advantageous in case of the availability of source symbols with exact probability distribution as compared to the case when source statistics are changing. Another effective variable length coding method is Golomb coding. In this work Golomb coding and two values Golomb coding are used for coding of indices.

III. INDEXING OF LATTICE VECTORS

In lattice vector quantizer design, a very important task is index assignment to the quantized vectors. Mainly two approaches are reported for calculating index of vectors one is based on lattice points enumeration and second based on leader's addressing. In enumerating lattice point approach the index consists of: leader's norm (n) and vector's index (i). In this indexing technique the total number of vectors lying on a given hypersurface is taken into account. In enumerating lattice point approach the

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication

index (n, i) consists of leader's norm (n) and vector's index (i). The non manageable value of the lattice vector lying on a hypersurface can speedily attain for practical execution [5][6] . Leader based indexing requires: leader's norm (n), vector's index (i) and vector's rank in the class of equivalence (r), so index consists of (n, i, r) values. These computations are not necessary in the projected method in [2]. Indexing method presented in this work is based on independent indexing of vectors, where complete vector is encoded into a scalar integer index. This method directly assigns indexes to the vectors. Following process is used to get index of the quantized vector (QV)

If (QV) = [v1 v2 v3 v4 v5 v6 v7 v8], index=(V1xbase7) + (V2xbase6) + (V3xbase5) + (V4xbase4) + (V5xbase3) + (V6xbase2) + (V7xbase) + (V8), where [V1 V2 V3 V4 V5 V6 V7 V8] = [v1 v2 v3 v4 v5 v6 v7 v8] + │smallest value of quantized vector (QV) │and base = largest value of quantized vector (QV) +│smallest value of quantized vector (QV)│ + 1

IV. GOLOMB CODING

In this section, we review the Golomb coding, two value Golomb coding and its application to binary data having long runs of zeros. Important parameters of Golomb coding algorithm are tunable parameter M called as group size, count of continuous number of 0's followed by 1 called runlength N. Selection of the parameter M is the first step in the encoding procedure. The value of M is taken through which the best compression is achieved. When the group size M is determined, the runs of zeros in precompiled data set are mapped to groups of size M (each group corresponding to a run length). The number of such groups is determined by the length of the longest run of zeros in the precompiled data set. The set of run lengths {0, 1, 2 . . ., M-1} form group G1; the set {M, M+1, M+2 . . . , 2M-1}, G2 group; etc. In general, the set of run lengths {(q-1) M, (q-1) M+1, (q-1) M+2 . . ., qM-1} comprises group Gq , where q denotes the group number. A group prefix is assigned to each group of (q - 1) 0s followed by a one. It is denoted by 0(q-1)1. If M is chosen to be a power of two, i.e., $M = 2^N$, each group contains 2^N members and a $\log_2 M$-bit sequence (tail) uniquely identifies each member within the group. Thus, the final code word for a run length L that belongs to group Gq is composed of two parts, a group prefix and a tail. The prefix is 0(q-1)1 and the tail is a sequence of $\log_2 M$ bits. The encoding process is illustrated in Table 1. for M = 4.

Table 1 Golomb Coding Process

Group	Run length	Group prefix (Quotient)	Tail (Remainder)	Codeword
G1	0	1	00	100
	1	1	01	101
	2	1	10	110
	3	1	11	111
G2	4	01	00	0100
	5	01	01	0101
	6	01	10	0110
	7	01	11	0111
G3	8	001	00	00100
	9	001	01	00101
	10	001	10	00110
	11	001	11	00111
G4	12	0001	00	000100
	13	0001	01	000101
	14	0001	10	000110
	15	0001	11	000111
…	…	…	…	…

V. TWO VALUE GOLOMB CODING (2-VALUE GOLOMB CODING)

As compare to Golomb Coding in this method run length calculated as count of continuous 0's or continuous 1's. The advantage of such modification in algorithm is that as both runs of 0's and 1's are considered hence no need to insert 1 at the end of sequence like in Golomb coding. This modified algorithm is known as 2-value Golomb coding since both runs of 0's and 1's are considering. Table 2 illustrates an example to represent 47 input bits (S) using Golomb coding (S1) and 2-value Golomb coding (S2). Here long sequence of 1's may be coded with few bits which is not in the case of Golomb coding.

Example: S=000000000000100000000001000011111111110 0000001(47 input bits)
S1={0000000000001 00000000001 00001 1 1 1 1 1 1 1 1 1 1 00000001}(Golomb coding)
S2={000000000000 1 0000000000 1 0000 11111111111 0000000 1} (2-value Golomb coding)

Table 2 Two-value Golomb coding

(Golomb coding)									Total bits
Runlength	12	1 0	4	10 times 0	7	-	-	-	
codeword	0 0 0 1 0 0	0 0 1 1 0	01 00	10 times 000	0 1 1 1				49
(2-value Golomb coding)									Total bits
Runlength	12	1	10	1	4	11	7	1	
codeword	000 10 0	101	00 11 0	1 0 1	0 1 0 0	0 0 1 1 1	0 1 1 1	10 1	33

VI. CODING RESULT

Input image is decomposed to five DWT levels, reshaped in a way so that it can quantized using quantization algorithm of D8 lattice. Then indexes are assigned according to the developed algorithm. The indexes are coded using two value golomb codes. First the index sequence is changed to binary sequence, and split into two parts, that is, one part consists of higher four bits and the other part consist of remaining lower bits.The compression is done only on the higher bits because it has more continuous ones and zeros. The amount of compression obtained is computed as follows- From the binary representation of indexes in Table 3 (a) it has been examined that lower order bits have small run length of 1's and 0's as compared to higher order bits. Table 4 shows the compression ratio achieved for different values of M and the value of M for best compression for barbara and boat images.

Compression Ratio $CR = \frac{Td-Te}{Td} \times 100$

Where Td =Total no. of bits required to code the indexes Te =Total no. of bits required to code the indexes using 2-value golomb coding

A. Coding of indexes using 2-v Golomb Coding

When bits are scanned columnwise large number of continuous 1's and 0's are there as compared to rowwise scanning. Table 3(a) shows indexes of boat image and their binary representation. Table 3(b) shows coding by 2-V Golomb Coding.

Table 3(a) Binary representation of indexes

Indexes of one subband of boat image	Binary representation	
	Higher bits	Lower bits
145	1 0 0 1	0 0 0 1
150	1 0 01	0 1 1 0
161	1 0 1 0	0 0 0 1
160	1 0 1 0	0 0 0 0
156	1 0 0 1	1 1 0 0
166	1 0 1 0	0 1 1 0
166	1 0 1 0	0 1 1 0
166	1 0 1 0	0 1 1 0

B. Encoding of higher order bits of Indexes

Table 3(b) Coding by 2-V Golomb Coding

S(columnwise bits)	11111111 0000000000 11 0 11111 00 1 000	32 bits
RL(run length)	8 10 2 1 5 2 1 3	
2-V GC	00100 00110 110 101 0101 110 101 111	29 bits

Table 4 Encoding of higher order bits of indexes

Image	T_d bits	M	T_e bits	Compression Ratio (%)	Best Compression (M)
Barbara	32x4=128	2	127	0.78	4
		4	125	2.34	
		8	147	-14.84	
		16	181	-41.4	
Boat	32x4=128	2	119	7	4
		4	118	7.8	
		8	134	-4.68	
		16	162	-26.56	

C. Comparison of coding only higher order bits over complete bits

It has been observed from data shown in Table 5 that when only higher order four bits are considered compression is achieved for values of M=2 ,4 and 16 for boat image but no compression is there when all bits are coded.

Table 5 Comparison of coding

Image	considering only higher order four bits				considering all bits			
	TD (bits)	M	TE (bits)	CR (%)	TD (bits)	M	TE (bits)	CR (%)
Boat	32x 4= 128	2	119	7.03	32x8 =256	2	287	No compression
		4	118	7.8		4	315	
		8	134	-4.6		8	383	
		16	162	-26.5		16	473	

VII. CONCLUSION

Through the indexing method used in this paper index can be assigned with less computation as compared to leader indexing method since there is no need to find leader norm, vector index and vector rank etc. In this paper, 2-value Golomb coding is used to encode indexes of lattice codevectors. 2-value Golomb coding gives better results as compare to the Golomb coding. Coding results show that when input data is scanned columnwise more number of continuous ones and zeros are there i.e. runlength is more, when encoded using 2-value Golomb coding, less number of bits are required as compare to rowwise scan. When the data is scanned coloumnwise it is found that frequency of changing 0 to 1 and 1 to 0 is more for lower bits, for higher bits runlengths are more, only higher bits are encoded. Best compression is found for specific band of barbara and boat images. The developed method can be used for hyperspectral image compression and video compression.

REFERENCES

1. 1. John H. Conway and N.J.A. Sloane (1983). IEEE Trans.Information Theory 29, 820.
2. 2. J. Moureaux, P. Loyer and M. Antonini(1998). IEEE Trans. Communication, 46, 1602 .
3. Patrick Rault and Christine Guillemot (2001). IEEE Trans. Multimedia,3, 395.
4. L. H. Fonteles and M. Antonini (2007)., Proc. of IEEE International Symposium on Information Technology, 241.
5. T. R. Fiser (1986). IEEE Trans. Infomation Theory, 32, 568.
6. Z. G. F. Chen and J. Villasenor , IEEE Trans. Information Theory, 43, 92 (1997).
7. AdrianaVasilache, 16th European Signal Processing Conference (EUSIPCO 2008), Lausanne, Switzerland, (2008).
8. Ludovic Gullimet, Yann Gaudeau, Said Moussaoui and Jean-Marie Moureaux, IEEE Trans. on Image Processing, 17, (2008).
9. Krüger, H., Geiser, B., Vary, P., Li, H. T., Zhang, D., IEEE International Conference on Acoustics, Speech and Signal Processing, Prague, Czech Republic, (2011).
10. R. R. Khandelwal, P. K. Purohit, S. K. Shriwastava, , International Journal of Remote Sensing and GIS, 1, 77 (2012).
11. D. Huffman, Proc. Inst. Radio Engg., 40,1098 (1952).
12. K. Sayood, Introduction to data compression (Elsevier Inc., 2006).
13. A. Moffat and R. N. Neal, ACM Trans. Information Syst., 16, 256 (1998).
14. S. Golomb, IEEE Trans. Information Theory, 12, 399 (1966).
15. Consultative Committee for Space Data Systems (CCSDS), Lossless data compression (Blue Book, 1997)
16. Malvar, H., Proc. Data Compression Conf. (DCC), IEEE Computer Society, 23(2006).Rice, R.F., Yeh, P.S., Miller, W.H., Proc. AIAA Computing in Aerospace Conf., San Diego (1993).
17. Rice, R.F., Yeh, P.S., Miller, W.H., Proc. AIAA Computing in Aerospace Conf., San Deigo (1993).
18. Salomon, D., Variable-length codes for data compression (Springer,2007)
19. Chandra, A.; Chakrabarty, K., IEEE Trans. Computer-Aided Design, 21, 715 (2002).
20. R. Yu, C. C. Ko, S. Rahardja, and X. Lin, Proc. IEEE Intl. Conf. on Acoust., Speech, and Signal Process., 4, 277 (2003).
21. Walter D. Leon-Salas, Sina Balkir, Khalid Sayood, and Michael W. Hoffman, IEEE Trans.on Circuits and Systems-II,53, 278 (2006).
22. K. Somasundaram and S. Domnic, IEEE Trans. on Multimedia, 9, 239 (2007).
23. Jiantao Zhou, Oscar C. Au, and Amanda Yannan Wu, ICASSP (2009).
24. V.Glory, S. Domnic, IEEE International Conference on Advances in Engineering Science and Management, (2012).
25. Jian-Jiun Ding, Hsin-Hui Chen, and Wei-Yi Wei, IEEE Trans. on Circuits and Systems for Video Technology, 23, 661 (2013).
26. Savita Nargundmath, Archana Nandibewool, International Conference on Advanced Nanomaterials & Emerging Engineering Technologies, organized by Sathyabama University, Chennai, India in association with DRDO, New Delhi, India, (2013).
27. Zoran H. Peric´, JelenaR. Nikolic´, Aleksandar V. Mosic´, Institution of Engineering and Technology (IET) Communications, 8, 372 (2014).
28. Matthew F. Brejza, Tao Wang, Wenbo Zhang, David Al-Khalili, Robert G. Maunder, Bashir M. Al-Hashimi, (Fellow, IEEE), And Lajos Hanzo, (Fellow, IEEE), IEEE Access Journal, 4, 7154 (2016), DOI:10.1109/ACCESS.2016.2584982.
29. N. J. A. Sloane, IEEE Trans. Information Theory, 27, (1981).
30. M., Sol´e, P., Gaidon, T., Antonini, M. and Mathieu, P., IEEE Trans. Image Processing, 3, 367 (1994).
31. Vasilache, A., Vasilache M. and Tabus, I., Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing, 2 (1999).
32. J. H. Conway and N.J.A. Sloane, IEEE Trans. Information Theory, 28, 227 (1982).

AUTHORS PROFILE

Richa R. Khandelwal, completed her graduation from Madhav Institute of Technology and Science (MITS), Gwalior (Madhya Pradesh) and post graduation in Electronics Engineering from Yeshwantrao Chavan College of Engineering (YCCE), Nagpur, (Maharashtra). She did her Ph. D. from Barkatullah University Bhopal. Her employment experience includes eighteen years of teaching at graduate/post graduate level. She is having to her credit many International and National Conference/Journal papers. Her special fields of interest include Image Processing.
She has guided UG and PG projects related with the application of image processing.

P.K. Purohit, did his M.Sc. (Physics) and Ph.D. in communication Electronics from Barkatullah University Bhopal. He is Professor in Department of Science, National Institute of Technical Teachers' Training and Research Bhopal (Under Ministry of Human Resource Development, Govt. of India).He has participated in 22nd Indian Antarctic Scientific Expedition. He has written three books on Antarctica. He has participated/ presented papers in National /International seminar/ conferences. He has published many papers/ articles in various Magazines/Journals/ Newspapers. He has been associated with National and International Research projects. He is guiding Students for Ph.D. degree in Physics and Electronics subject. For last 10 years his students are participating in Indian Antarctic Scientific programmer. He has written 06 books.