International Journal of Advanced Research in

Computer Science Engineering and Information Technology

Volume: 1 Issue: 1 06-Jun-2013,ISSN_NO: 2321-3337

Efficient Architecture for SPIHT Algorithm in Image Compression

R. VANAJA[1], N. LAKSHMI PRABA[2], DR. N. STALIN[3]

M.E (Communication Systems), Dhanalakshmi Srinivasan Engg. College[1],

Research Scholar, Anna University, Chennai[2],

Department of Petrochemical technology, Anna University of Techn., BIT Campus, Trichy[3],

vanaja2107@gmail.com[1], laksh_abi@rediffmail.com[2], mnstalin@gmail.com[3]

*ABSTRACT-**A arithmetic coder architecture with a high throughput memory efficient for set partitioning in hierarchical trees is proposed in this paper. This paper also presents a throughput efficient image compression using 'Set Partitioning in Hierarchical Trees' (SPIHT) algorithm for compression of images. The SPIHT use inherent redundancy among wavelet coefficients and suited for both gray and color image. The SPIHT algorithm uses dynamic data structures which hinders hardware realizations. In this FPGA implementation have modified basic SPIHT in two ways, one by using static (fixed) mappings which represent significant information and the other by interchanging the sorting and refinement passes. A hardware realizations is done in a Xilinx XC3S200 device. As one part of the hardware realization, the SPIHT algorithm was implemented in software side. In this matlab GUI (Graphical User Interface), the various images are compressed and is implemented without affecting the original quality of the image. The SPIHT algorithm can be applied to both grey-scale and colored images. Comparison of SPIHT in both the arithmetic coder and pipelined architecture was enumerated in this paper. SPIHT displays exceptional characteristics over several properties like good images quality, fast coding and decoding, a fully progressive bit stream, application in lossless compressions, error protection and ability to code for exact bit rate.***

***Key words*-SPIHT, arithmetic coding, DWT, compression process, wavelet transform, encoder and decoder, offsprings, list of insignificant pixels (LIP), list of significant pixels (LSP), list of insignificant sets(LIS), VLSI arithmetic coder architecture.**

I. INTRODUCTION

The arithmetic coding[2], [3] method was widely used by various image compression algorithms, since it has an ability to optimal performance i.e., to generate codes with fractional bits. The algorithms using the arithmetic coding method are QM in JPEG[4], MQ in JPEG 2000 and the context based adaptive binary arithmetic coder in H.264[5], [6]. Actually, the arithmetic coding method was used by the SPIHT algorithm to improve its peak signal to noise ratio. For some of the real time applications like satellite image compression and high speed camera image compression, the arithmetic coding method has limited its applications since, the theory and program code of arithmetic method are complicated. In the compression scenarios, the high speed architecture of arithmetic coding method should be designed to achieve performance gain and also to meet the throughput requirements.

For various image compression systems, the efforts has been put to arithmetic coder hardware architectures by industrial and academic research groups. For designing the high speed applications there are two important challenges. The first one is, as the arrival of many data, increased greater precision is required by arithmetic coder. The second is, during the adaptive model update and internal loops, the arithmetic coding requires the result of iteration i before next run. For the oast years, various architectures are proposed in order to deal with such difficulties.

A systolic hardware architecture was proposed by Wiseman [9], it is a simple version of arithmetic coder. In each stage of arithmetic coder a pipeline processing is used by the architecture [9]. Since [9] uses the pipeline processing, a fast lookup table is utilized and also the high frequency clock is eliminated. Andra [11] proposed QM coder in JPEG. In [11] architecture, the more probable symbol has been decreased and also for the speed up purpose it uses a non-overlap window. In order to simplify renormalization operation, the non-overlap window is applied to the continuous more probable symbol in [11]. There is decrement in the performance by 1% to 3% in Andra's coder, hence the operations are simple.

In order to improve the performance of SPIHT, various modifications are made by many researchers. Some of the algorithms are not even thinking about the hardware issues, they mainly aim for better PSNR values. For selecting the wavelet packet transform a method was introduced by Kassim [12]. In [12], the conflict of parental relationship was avoided and also compresses the high frequency subband energy into few trees efficiently. For highly textured images, the coding gains was improved and achieved by SPIHT wavelet packet transform. A context based SPIHT method was introduced by Ansari [13]. In [13], it achieves a better results in medical images and its performance too good, since it uses segmentation and possible method for selecting the region of mask. A method was proposed by Akter [14] to reduce memory and to speed up ASPIHT software. In [14], Akter used to store the coordinates of wavelet coefficients in one list of SPIHT instead of three lists of SPIHT. Without using lists a modified algorithm for SPIHT was proposed by Wheeler [17]. The algorithm speed has been improved since there is no insertion operation and search operations for lists.

II. RELATED WORK

Related ideas have recently been explored in the literature. The architectures of arithmetic coder used in

SPIHT [1], provides the arithmetic coder's core structure and the internal structure of arithmetic coder. During implementation, there are three stages for the architecture. The first stage is, the SPIHT using arithmetic coder has been designed which is suitable for hardware processing using context model. This context model is designed by Fowler, in the QccPack SPIHT. In Fowler architecture, the QccPack SPIHT software was designed for the context model. It uses the four different context and also exploits the relationship of modes. The four different context are used for current position value, current position sign, descendant set and grant descendant set. The second stage is, arranging modules for hardware and removing the internal loops of arithmetic coder. The third stage is, to build a arithmetic coder all the modules are connected together using different paths. The arithmetic coding [3], obtains optimal performance for its ability to generate codes with fractional bits. The arithmetics coding for data compression, [15] by I. C. Witten, R. M. Neal and J.G. Cleary. They clearly gives the arithmetic coding architecture for the data compression.

III. BACKGROUND AND MOTIVATION

A.Principle of Arithmetic Coding

The set of symbols be L. Each symbol i L. Its probability is given as P_i [0,1]. Let X^M be the codeword for symbol sequence i.e., $X^M = \{X_1, X_2 .... X_M\}$. For X^M, the $P(X^M)$ is the probability and $q(X^M)$ is the cumulative probability. i.e., $(X^M) = [q(X^M) + p(X^M)] /2$. When the new symbols are arrived, the coding interval length will be decreased by arithmetic coder. The coding interval shrink will be slow, when the input probability is high. In practical, the performance of conditional probability is high than non- conditional probability. In various fields the context based arithmetic coder is mostly used, context insense, conditions for current symbol. But the context refers to the neighbour pixels states in the image coding. The core structure of Arithmetic Coder was shown in Fig. 1.

IV. PROPOSED WORK

One of the major challenges in enabling mobile multimedia data services will be the need to process and wirelessly transmit a very large volume of data. While significant improvement in achievable bandwidth are expected with future wireless access technologies, improvement in battery technology will lag the rapidly growing energy requirements of future wireless data services. One approach to mitigates to this problem is to reduce the volume of multimedia data transmitted over the wireless channel via data compression techniques.

A. Image Compression. Compression is a process of reducing the number of data bits necessary for representing a information, to properly utilizes the available bandwidth and reduce storage spaces. There are two types of compression namely Lossless compression and Lossy compression.

B. Lossless Compression. In lossless compression data can be completely recovered after decompression. Recovered data is identicals to original, exploits redundancy in data

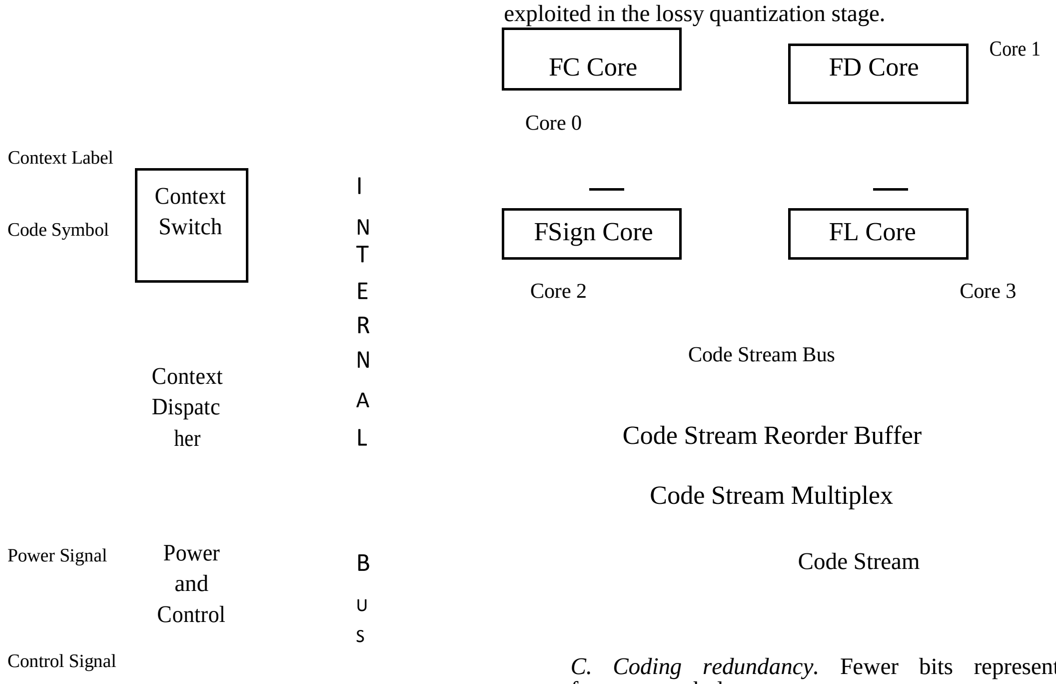

Fig. 1. Arithmetic Coder's Core Structure

C. Lossy Compression. Data cannot be recovered after decompression, some informations is lost forever, gives more compressions than lossless, discards "insignificant" data components.

Principle of compression

Image compression addresses the problem of reducing the amount of data required to represent a digital image. The underlying basis removals of redundant data. The transformations storage and transmissions. The compressed image is decompressed at some later time, to reconstruct the original images or an approximation to it.

Different types of data redundancies

A. Interpixel redundancy. Neighboring pixels have similar value. This property is exploited in the wavelet transform stage.

B. *Psychovisual redundancy.* Human visual system cannot simultaneously distinguish all color. This property is exploited in the lossy quantization stage.

C. Coding redundancy. Fewer bits represent frequent symbols.

Image Compression Process

The image sample first goes through a transform, which generate a set of frequency coefficients. The transformed coefficient are then quantized to reduce the volume of data. The output of this steps is a stream of integers, each of which correspond to an index of particular

quantized binary. Encoding is the final step, where the stream of quantized data is converted into a sequence of binary symbols in which shorter binary symbols are used to encode integers that occur with relatively high probability. This reduce the number of bits transmitted. This is illustrated in Fig. 2.

Wavelet image compression

The foremost goal is to attain the best compression performance possible for a wide range of image classes while minimizing the computational and implementation complexity of the algorithms. For a compression algorithm to be widely useful, it must performs well on a wide variety of image content while maintaining a practical compression/ decompression time on modest computer. In order to allow a broad range of implementation, an algorithms must be amenable to both software and hardware implementation.

Typical image coder

B. Discrete Wavelet transform

A typical image compression system consisting of three closely connected components namely (a) Source Encoder (b) Quantizer and (c) Entropy Encoder is shown in Fig. 3. Compression is accomplished by applying a linear transform to decorrelate the images data, quantizing the resulting transform coefficients and entropy coding the quantized values.

The source coder decorrelate the pixels. A variety of linear transforms have been developed which include Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete wavelet Transform (DWT) and many more, each with its own advantage and disadvantages.

A quantizer simply reduces the number of bits needed to store the transformed coefficients by reducing the precision of those value. Since this is a many-to-one mapping, it is a lossy processes. Quantization can b performed on each individual coefficient, which is known as *Scalar* Quantization (SQ). Quantization can also be performed on a group of coefficients together, and this is knowns as *Vector* Quantization (VQ).

Wavelet transform overview

A. Wavelet transform. Wavelets are mathematical functions defined over a finite interval and having an average value of zero that transform data into different frequency component, representing each component with a resolution matched to its scale.

The basic idea of the wavelet transform is to represent any arbitrary function as a superposition of a set of such wavelets are basis functions. These basis functions are baby wavelets are obtained from a single prototype wavelet called the mother wavelet, by dilations or contractions (scaling) and translations (shifts).

Calculating wavelet coefficients at every possible scale is a fair amount of work, and it generates an awful lot of datas. If the scales and positions are chosen based on power of two, the so called dyadic scales and position, then calculating wavelet coefficients are efficient and just as accurates. This is obtained from discrete wavelet transform (DWT).

C. SPIHT algorithm

Let's briefly describe the algorithms. However, before we do that we need to get familiar with some notation. The data structures used by the SPIHT algorithm is similar to that used by the EZW (Embedded Zerotree Wavelet) algorithm-although not the same.

The wavelet coefficients are again divided into trees originating from the lowest resolution band (band I in our case). The coefficient are grouped into 2 × 2 arrays that, except for the coefficient in band I, are offsprings of a coefficient of a lower resolutions band. The coefficients in the lowest resolution band are also divided into 2 × 2 arrays. The coefficient in the top-left corner of the array does not have any offsprings. The data structure is shown pictorially in Fig. 5, for a seven-band decomposition.

The trees are further partitioned into four types of sets, which are sets of coordinates of the coefficients. $\mathcal{O}$(i,j). This is the set of coordinates of the offsprings of the wavelet coefficient at $\mathcal{O}$(i,j) is either location (i,j). As each node can either have four offsprings or none, the size of zero or four. For example, in Fig. 5, the set $\mathcal{O}$(i,j) consists of the coordinates of the coefficients b_1, b_2, b_3 and b_4. $\mathcal{D}$(i,j).

This is the set of all descendents of the coefficient at location (i,j). Descendants include the offsprings, the offsprings of the offsprings, and so on. For example, in Fig.5, the set $\mathcal{D}$(i,j) consists of the coordinates of the coefficients b_1,…., b_4, b_{11},…., b_{14}, …., b_{44}. Because the number of offsprings can either be zero or four, the size of $\mathcal{D}$(i,j) is either zero or a sum of powers of four.

International Journal of Advanced Research in

Computer Science Engineering and Information Technology

Volume: 1 Issue: 1 06-Jun-2013,ISSN_NO: 2321-3337

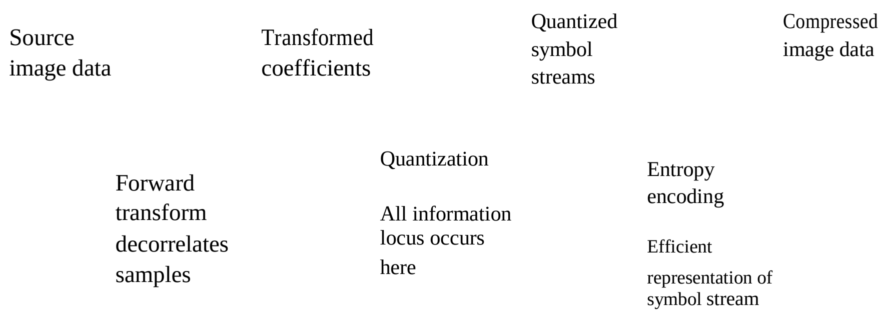

Fig. 2. Bock diagram of image compression process.

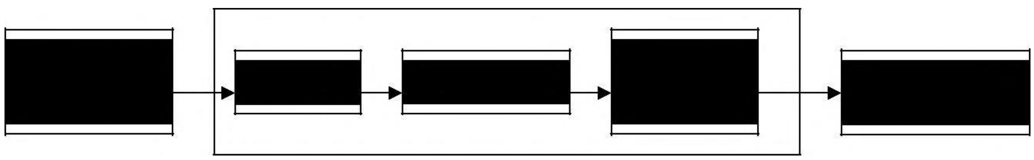

Fig. 3. Wavelet Encoder.

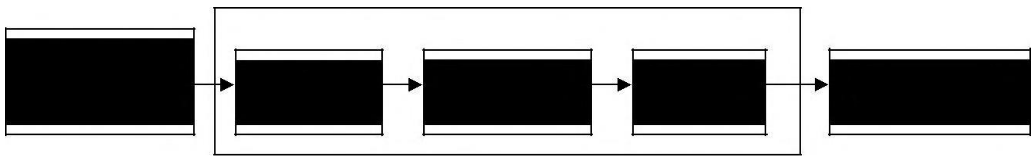

Fig. 4. Wavelet Decoder.

$\mathcal{H}$ This is the set of all root nodes- essentially band I in the case of Fig. 5.

$\mathcal{L}$(i,j) This is the set of coordinates of all the descendants of the coefficient at location (i,j) except for the immediate offsprings of the coefficient at location (i,j). In other words,

$$\mathcal{L}\ (i,j) = \mathcal{D}\ (i,j) - \mathcal{O}\ (i,j).$$

The algorithm makes use of three lists: the list of insignificant pixels (LIP), the list of significant pixels (LSP) and the list of significant sets (LIS). The LSP and LIS lists will contain the coordinates of coefficients, while the LIS will contain the coordinates of the roots of sets of type $\mathcal{D}$ or $\mathcal{L}$. We star by determining the initial value of threshold. We do this by calculating,

$$n = |\ \log2\ c_{max}|$$

where C_{max} is the maximum magnitude of the coefficients to be encoded. The LIP list is initialized with the set $\mathcal{H}$.

Those elements of $\mathcal{H}$ that have descendants are also placed in LSP as type $\mathcal{D}$ entries. The LSP list is initially empty.

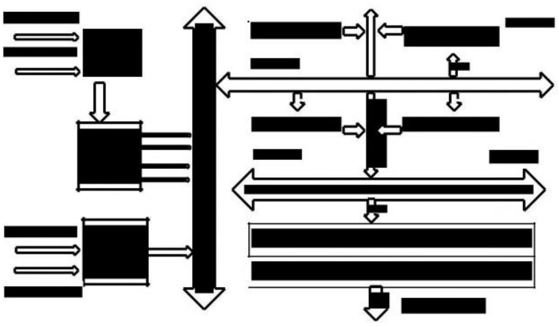

Fig. 5. Data structure used in the SPIHT algorithm.

TABLE I

PSNR RESULTS

Image	SPIHT-AC	Simulated result from Matlab GUI
Lena	40.41	-
Barbara	36.41	-
Airport	33.27	Inf
Pentagon	34.31	-
Café	31.71	-
Normal image taken from digital camera	37.70	Inf
A medical image	38.28	Inf

The PSNR value for various images using Arithmetic coder and matlab GUI was illustrated in Table 1. If the PSNR value is high then the quality of the image will be good and better. From Table 1 it is clear that, the image obtained from simulation using matlab GUI has better image quality exactly as the original image. But the compressed image obtained from arithmetic coder has lower image quality or poor image quality than that obtained from simulation.

V. SIMULATION RESULTS

In the section we analyze the performance of SPIHT algorithm through simulations based on MATLAB GUI tool. In the simulation, the original image was compressed to give the reconstructed image and the CR, PSNR, MSE values are shown below.

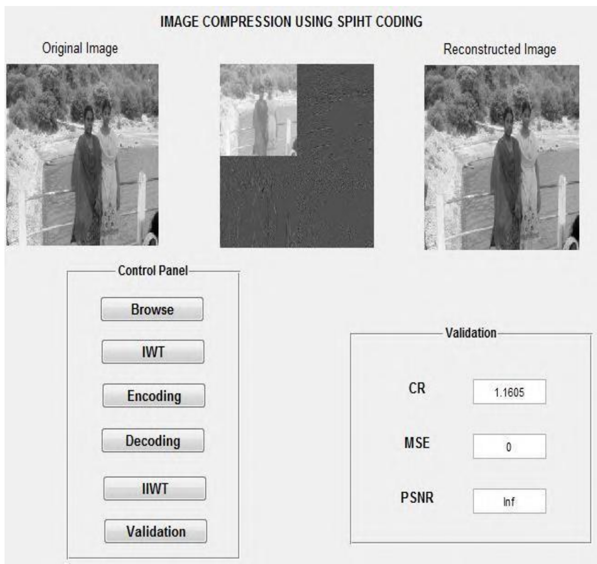

Fig. 6. Image compression using SPIHT algorithm for a normal image.

The normal image gives the results of compression ratio as 1.605, MSE value as zero and the PSNR value as infinity. This simulated result was shown in Fig. 6. While the simulated results for using medical field image are given as, the compression ratio 1.44, MSE value as zero and the PSNR value is infinity. This result was simulated and shown in the Fig. 7. Further the same result was implemented in FPGA kit, using the pipelined architecture for the SPIHT algorithm in the future work.

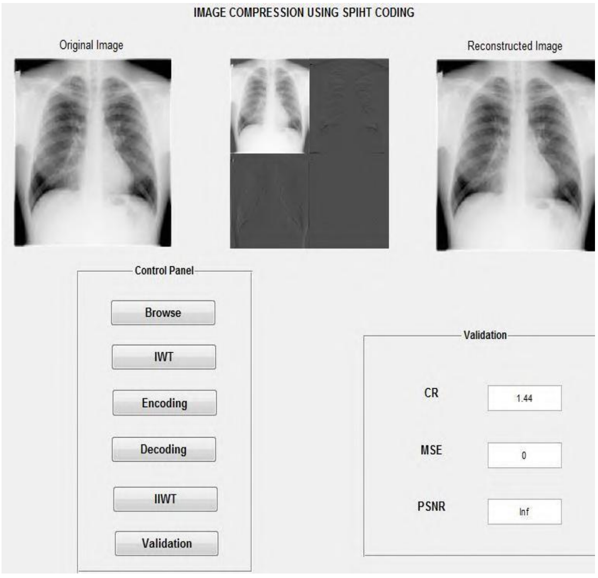

Fig. 7. Image compression using SPIHT algorithm for a medical field image.

VI. CONCLUSION

This paper brings the compression processes, the wavelet transform and the wavelet coefficients. The data redundancies and the types of data redundancies are also discussed. Also the SPIHT algorithm was explained in detail.

SPIHT is a wavelet based compression. Since the discrete wavelet transform plays a major role in the SPIHT algorithm This paper also enumerates, how the datas are partitioned in the SPIHT algorithm. The thresholds used for checking significance are powers of two, so in essence the SPIHT algorithm sends the binary representation of the integer value of the wavelet coefficients.

REFERENCES

[1] Kai Liu, "VLSI Architecture of Arithmetic Coder Used in SPIHT" ,Evgeniy Belyaev, Jie Guo., vol.20, no.4, April 2012.

[2] J. Rissanen. " Generalized kraft inequality and arithmetic coding", IBM J. Res. Developm., vol. 20. No. 3, pp. 198-203, may 1976.

[3] J. Rissanen and G. G. Langdon, " Arithmetic coding", IBM J. Res. Developm., vol. 23, no. 2, pp. 149-162, mar. 1979.

[4] ISO/IEC JTCI Information technology-Digital Compression and Coding of Continuous-Tone Still Images-PartI: Requirements and Guidelines, ISO/IEC International Standard 10918-1, ITU-T Rec. T.81, 1993.

[5] Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec, H.264 ISO/IEC 14496-10 AVC), JTC-G050, Joint Video Team ITU-T and ISO/IEC JTC 1, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, Mar. 2003.

[6] D. Marpe, H. Schwarz and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard", IEEE Trans. Circuits Syst. For Video Technol., vol. 13, no. 7, pp. 620-636, Jul. 2003.

[7] B. Cao, Y.-S. Li and K. Liu, " VLSI architecture of MQ encoder in JPEC2000", J. Xidian Xuebao, vol. 31, no. 5, pp. 714-718, Oct. 2004.

[8] A. Said and W. A. Pearlman, " A new, fast ans efficient image codec based on set partitioning in hierarchical trees", IEEE Trans. Circuits Syst. For Video Technol., vol. 6, no. 3, pp. 243-249, Mar. 1996.

[9] Y. Wiseman, " A pipeline chip for quasi arithmetic coding", IEICE Trans. Fundamentals, vol. E84-A, no.4, pp. 1034-1041, Apr. 2001.

[10] J. Jyotheswar and S. Mahapatra, " Efficient FPGA implementation of DWT and modified SPIHT for lossless image compression", J. Syst. Arch., vol. 53, no. 7, pp. 369-378, Jul. 2007.

[11] K. Andra, T. Acharya and C. Chakrabathi, " A multi-bit binary arithmetic coding technique", in Proc. Int. Conf. Image Process., Vancouver, BC, Canada, Sep. 2000, vol. 1, pp. 928-931.

[12] A. A. KAssim, N. Yan and D. Zonoobi, " Wavelet packet transform basis selection method for set partitioning in hierarchical trees", J. Electron. Imag., vol. 17, no. 3, p. 033007, Jul. 2008.

[13] M. A. Ansari and R. S. Ananda, " Context based medical image compression for ultrasound images with contextual set partitioning in hierarchical trees algorithm", Adv. Eng. Softw., vocl. 40. no. 7, pp. 487-496, Jul. 2009.

[14] M. Akter, M. B. I. Reaz, F. Mohd-Yasin and F. Choong, " A modified-set partitioning in hierarchical trees algorithm for real-time image compressions", J. Commun. Technol, Electrons., vol. 53, no. 6, pp. 642-650, Jun. 2008.

[15] I. C. Witten, R. M. Neal and J. G. Cleary, " Arithmetic coding for data compression", Commun. ACM, vol. 30, no. 6, pp. 520-540, Jun. 1987.

[16] M. Dyer, D. Taubman and S. Nooshabadi, " Concurrency techniques for arithmetic coding in JPEG2000", IEEE Trans. Circuits Systems I,Reg. Papers, vol. 53, no. 6, pp. 1203-1213, jun. 2006.

[17] F. W. Wheeler and W. A. Pearlman, "SPIHT image compression without lists," in Proc. IEEE Int. Conf. Acoust., Speech, Signal process., Istanbul, Turkey, Jun. 2000, pp. 2047-2050.

Biography

Mrs. N. Lakshmi Praba doing PhD in Anna University of technology, Trichy. Did her Master of Technology (M.E) in Shanmuga college of Engineering, Tanjore and completed her Bachelor of Engineering (B.E) in Noorul Islam college of Engineering, Kumaracoil, Kanyakumari District.

Miss. R. Vanaja doing Master of Engineering (M.E), Communication Systems in Dhanalakshmi Srinivasan Engineering college, Perambalur. Completed her Bachelor of Engineering (B.E), Electronics & Communication Engineering in Thirumalai Engineering college, Kanchipuram.