ADAPTIVE MESH BASED 3D MR IMAGE COMPRESSION USING JPEG CODING

Bhavani .S[1], **Dr.Thanushkodi K.** [2]

[1,2]Akshaya College of Engineering and Technology, Coimbatore
[1]Research Scholar, Anna University, Coimbatore
Email: [1]bhavanisns@yahoo.com

Abstract

The MR images play a major role in the diagnosis of vital organs of the human body. Huge amount of medical image data is generated on a daily basis. This data needs to be stored for future study and follow up. This requires a large amount of storage space which is especially true for three - dimensional (3-D) medical data formed by image sequences. This has resulted in image compression being an important issue in reducing the cost of data storage and transmission time. Mesh based compression is one among them. JPEG is the currently accepted industry standard for still image compression. This paper explains the compression based on content-based non-uniform meshes, spatial transformation for motion compensation of MR images using JPEG which optimizes the above said issue and also aids in telemedicine applications. Compression rate achieved in 3 D wavelet scheme is 1.86(8), Multi dimensional Layered Zero Coding is 2.16(8) and in our coding using JPEG it is found to be 2.64.

Key words: Content-based mesh, Context-based modeling, Image Compression, Motion Compensation, JPEG coding.

I. INTRODUCTION

Recent advances in medical imaging technology have led to ubiquitous use of medical images for diagnosis. The major application of imaging in radio band is in medicine and astronomy. In medicine radio waves are used in magnetic resonance imaging. This technique places a patient in a powerful magnet and passes radio waves through his or her body in short pulses. Each pulse causes a responding pulse of radio waves to be emitted by the patient's tissues. The location from where these signals originate and their strength is determined by computer which produces 2-D picture of the section of the patient.

The 3-D medical image data used for diagnosis of vital organs like heart and brain employs Magnetic Resonance (MR) images (3).The limitations in transmission bandwidth on one side and the growing size of the medical image datasets on the other push towards the design of new design tools for efficient implementation With the recent developments in tele-medicine, there is also a need for reduction in transmission time. The increasing demand has triggered a vast investigation on compression of volumetric data sets and the solutions proposed so far has exploited the dependencies among 3D space. Among these some are concerned with video sequences while others are focused mainly on medical data (2)-(3),(5)

This paper is organized as follows. Section II presents overview of JPEG and its applications. In Section III the preprocessing and adaptive mesh formation of the image is briefed out. In Section IV the motion compensation involved in the design is explained. Section V illustrates the Implementation Plan and the procedure of JPEG coding. In Section VI Results and discussions are summarized and Section VII derives conclusion.

II. JPEG OVERVIEW

JPEG is a standardized image compression mechanism. JPEG stands for Joint Photographic Experts Group. It is designed for compressing full-color or gray-scale images of "natural", real world scenes. The algorithm achieves much of its compression by exploiting known limitations of the human eye, notably of the fact that small color details aren't perceived as well as small details of light-and-dark.

JPEG itself specifies both the codec, which defines how an image is compressed into a stream of bytes and decompressed back into an image, and the file format used to contain that stream. The most common file extension for this format is .jpg, though .jpeg, .jpe, .jfif, and .jif are also used.

Some of the advantages of JPEG are

1. To make the file smaller and having them appear virtually the same as the original.

2. JPEG is capable of storing full color information 24 bits/pixel, (16 million colors).

3. JPEG take up less space on a memory card and are ready to use straight from the camera.

III PREPROCESSINGAND ADAPTIVE CONTENT BASED MESH FORMATION

Pre-Processing images prior to encoding, is done to remove noise or unimportant details from the image, to improve the overall performance of an image coder. In this paper we undergo image preprocessing using Morphological operations to obtain the Noise free background of the image.MRI consists of two parts the air part and the flesh part. Air part does not contain any clinical information. It is just a noise. It consumes unnecessary bits and impairs the performance of compression whereas the flesh part contains useful clinical information to be compressed without any loss. We go for masking the image such that the total flesh part is included but the air part is coded zero. Image masking is done using Morphological operations where we code flesh part as “One” and air part as “Zero”. The steps involved in preprocessing are listed below.

(i) Binarize the image with a suitable threshold obtained from the histogram of the image.

(ii) In case if the holes are formed in foreground region of the binary mask, close these holes by binary morphological “Closing” operation.

(iii) Background part of the binary mask may contain spurious lines. Use Binary Morphological “Erode” operation to remove these lines.

(iv) Finally the mask contains the value of “1”corresponding to the Flesh part and a value of “0” to the Air part. Then the original image is multiplied with the resulting binary mask to suppress the background, keeping the information in fore-ground.

Fig. 4. and Fig. 5. shows the original and the noise free image. In mesh-based schemes the image is divided into representative elements and the deformation of each element in the subsequent frame is modeled by a spatial transformation. Triangular elements are used for mesh generation. The multiple deformations within a mesh element are avoided by generating the mesh based on the content of the image. The basic idea is to place the node points in such a way that mesh boundaries align with the object boundaries and the density of node points is proportional to the local deformation. In the content based mesh design we find the density of the elements by placing nodes at the edges. The density of mesh elements is decided from the optical flow of the region computed using the method of Horn-Chunk and it should be high in the regions of high optical flow.

Let I_k be the current frame and $I_k + 1$ be the next frame. The mesh is to be generated on I_k by taking its spatial transformation and optical flow between I_k and $I_k + 1$. Let DFD (x, y) be the displaced frame difference, computed as

$$DFD\,(x, y) = I_k\,(x, y) - I_k\,(x, y) \qquad \text{... (1)}$$

Where $I_k\,(x, y) = I_k + 1\,(x - \sigma x, y - \sigma y)$ is an estimation of $I_k\,(x, y)$ based on the optical flow vector $(\sigma x, \sigma y)$

A. Procedure For Mesh Generation

1. Label all pixels as “unmarked.”

2. Compute the average displaced frame difference $\mathrm{DFD_{avg}}$ as

$$\mathrm{DFD_{avg}} = \Sigma\,(\mathrm{x, y})\mathrm{DFD}^2(\mathrm{x, y})/\mathrm{K} \qquad \text{... (2)}$$

where K is the number of unmarked pixels and DFD is displaced frame difference.

3. Find the spatial edge map of the image using the “Canny” edge detector.

4 Select a pixel as a “node” if it is “unmarked”, falls on a spatial edge and is sufficiently away from all the previously marked pixels.

5 Grow a circle around this node point until in this circle is greater than DFD^2 (x, y). Label all pixels within the circle as “marked.”

6. Go to Step 2. until the required number of nodes is selected or the distance criterion is violated.

7. Given the selected nodes, apply a Delaunay triangulation to obtain the required content-based mesh.

The algorithm used finally ensures that the nodes are placed along the edges and the density of the elements is decided by the amount of activity between the two frames which is given by DFD. Fig. 6. shows the formation of Adaptive mesh based on optical flow.

IV. MOTION COMPENSATION

Motion Compensation (7) is a technique for describing a picture in terms of translated copies of portions of a reference picture, often 8×8 or 16×16-pixel blocks. This increases compression ratios by making better use of redundant information between successive frames.

The final image for the compression is synthesized using the spatial transformation. The mesh connectivity in the image is ensured using the block matching algorithm and the pixels corresponding to horizontal and vertical edges are determined. In geometry, an affine transformation between two vector spaces (strictly speaking, two affine spaces) consists of a linear transformation followed by a translation:

$$\mathrm{X} \to \mathrm{Ax} + \mathrm{B} \qquad \text{... (3)}$$

In the finite-dimensional case each affine transformation is given by a matrix A and vector b, which can be written as the matrix A with an extra column b and the image after affine transformation is shown in the Fig. 7 .The pixel values in smooth regions can be predicted from the previous frame using the affine transformation.

The residue is calculated using entropy coding without any loss.

$$r(u, v) = I_{k+1}(u, v) - \hat{I}_{k+1}(u, v) \qquad \text{... (4)}$$

Context-based entropy coders have gained prominence in achieving lossless compression of images. If '*X*' represents the random field of the residue to be coded and '*C*' denotes the context (or conditioning class) associated with each element of '*X*', then one can show that the conditional entropy $H(X/C)$ is less than or equal to the entropy $H(X)$. The reduction in entropy can be achieved by carefully choosing the contexts.

In this paper we choose 3-D contexts for entropy coding of source symbols. Two different contexts, namely, an energy context and a texture context are considered and the compound context is formed by combining energy, texture context with the residue. The final residues are taken for JPEG compression.

The prediction error depends upon the smoothness of the image around the predicted pixel and the displacement of the pixel. The error energy is estimated as,

$$\Delta = ad_h + bd_v + cm_v + d\,|\,e_w\,| \qquad \text{... (5)}$$

Where, $e_w = I_{k+1}(x, y) - \hat{I}_{k+1}(x, y)$, d_h and d_v denote the estimations of the horizontal and vertical edges, and m_v denotes the displacement of the pixel (x, y). The previous prediction error e_w is included because large errors tend to occur consecutively. d_h and d_v are determined from the already coded pixels.

The coefficients a, b, c and d are determined off-line using the least squares method, employing two consecutive slices of our data set. The obtained values are $a = 1.27$, $b = 1.2$, $c = 0.25$, and $d = 0.02$. Now, the residue $r(x, y)$ is conditioned on Δ so that the prediction errors are classified into classes of different variances. Since a large number of conditioning classes lead to the problem of context dilution, we quantize Δ into eight levels using the Lloyd-Max quantizer. By experimenting with other slices in the two data sets, we found that the values of a, b, c, d vary very little; also, these variations do not affect the quantization levels of Δ and, hence, the coding performance. Therefore, we use the above values for both the data sets.

We form a texture context C that nearly surrounds the pixel by using four causal neighbors in the current frame and four neighbors in the previous frame. This context captures higher order image patterns like texture patterns. We quantize Q vector into an 8-ary binary number C, by thresholding as follows, where p is any neighbor:

$$C(p) = \begin{cases} 0 & \text{if } Q(p) \geq \hat{I}_{k+1}(x\,y) \\ 1 & \text{otherwise} \end{cases} \qquad \text{... (6)}$$

A compound context is formed by combining the texture context and four levels of the energy context along with the residue. This can be viewed as a product quantization of two independent image features. The final residue obtained as shown in the Fig. 8. is given as,

$$\text{Final residue} = \mathrm{C}\,(\mathrm{p})^{*}\,\Delta^{*}\mathrm{r}\,(\mathrm{x},\,\mathrm{y}) \qquad \text{... (7)}$$

V. IMPLEMENTATION PLAN

Arithmetic coding of the residues obtained from adaptive mesh based context modeling already exists. Based on the approach mentioned in (7) the JPEG coding of the residues obtained from non uniform mesh is done and we found the compression ratio to be better than that obtained in (8).A comparison on the performance after region of interest coding is proposed to be studied.

B. JPEG Coding

For still image compression, the 'Joint Photographic Experts Group' or JPEG standard has been established by ISO (International Standards Organization) and IEC (International Electro-Technical Commission). The performance of these coders generally degrades at low bit-rates mainly because of the underlying block-based Discrete Cosine Transform (DCT) scheme.

In predictive coding, information already sent or available is used to predict future values, and the difference is coded. Since this is done in the image or spatial domain, it is relatively simple to implement and is readily adapted to local image characteristics. Differential Pulse Code Modulation (DPCM) is one particular example of predictive coding.

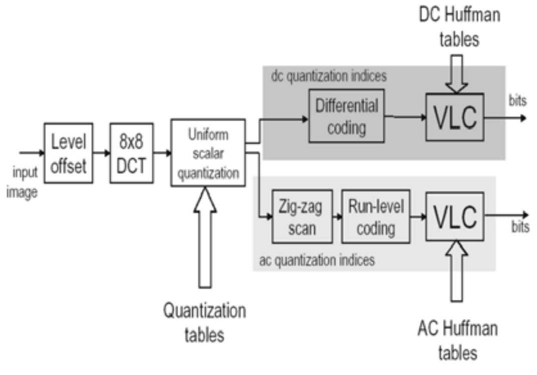

Fig. 1. Block diagram of JPEG encoder

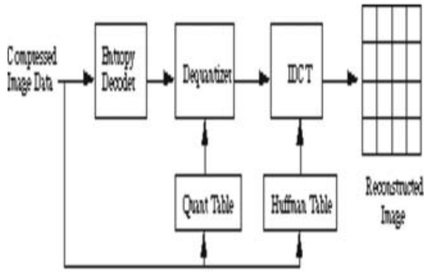

Fig. 2. Block diagram of JPEG decoder

Fig. 1. and Fig. 2. describes the JPEG encoding and decoding processes. Encoding of an image in JPEG is being done in JFIF format. The input of 24 bits per pixel is encoded and it's ready for further process. This particular block output is lossy data compression. Due to the densities of color- and brightness-sensitive receptors in the human eye, humans can see considerably more fine detail in the brightness of an image (the Y component) than in the color of an image (the C_b and C_r components). Using, encoders it is designed to compress images more efficiently. The ratios at which the down sampling can be done on JPEG are 4:4:4 (no down sampling), 4:2:2 (reduce by factor of 2 in horizontal direction), and most commonly 4:2:0 (reduce by factor of 2 in horizontal and vertical directions).After sub sampling, each channel must be split into 8'8 blocks (of pixels). If the data for a channel does not represent an integer number of blocks then the encoder must fill the remaining area of the incomplete blocks with some form of dummy data:

(i) Filling the edge pixels with a fixed color (typically black) creates dark artifacts along the visible part of the border.

(ii) Repeating the edge pixels is a common but non-optimal technique that avoids the visible border, but it still creates artifacts with the colorimetric of the filled cells.

(iii) A better strategy is to fill pixels using colors that preserve the DCT coefficients of the visible pixels, at least for the low frequency ones (for example filling with the average color of the visible part will preserve the first DC coefficient, but best fitting the next two AC coefficients will produce much better results with less visible 8'8 cell edges along the border).

Next, each component (Y, C_b, C_r) of each 8×8 block is converted to a frequency-domain representation, using a normalized, two-dimensional type-II discrete cosine transform (DCT). Before computing the DCT of the sub image, its grey values are shifted from a positive range to one centered around zero. For an 8-bit image each pixel has 256 possible values: [0,255]. To center on zero it is necessary to subtract by half the number of possible values, or 128.The next step is to take the two-dimensional DCT.

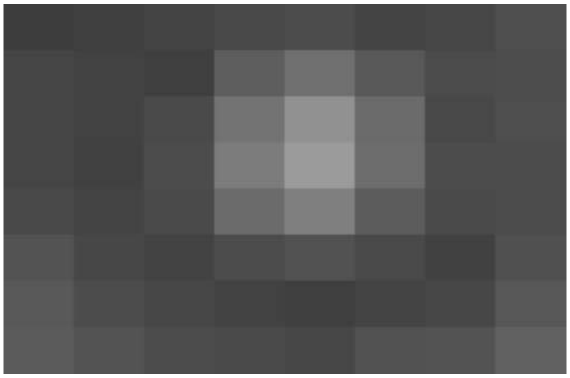

Fig. 3. 8 × 8 Sub image in 8-bit gray scale

The advantage of the DCT is its tendency to aggregate most of the signal in one corner of the result. The quantization step to follow accentuates this effect, while simultaneously reducing the size of the DCT coefficients to 8 bits or less, resulting in a signal with a large trailing region containing zeros so that the entropy stage can simply throw away.

The quantized DCT coefficients are computed using

$$B_j,\, k = round(A_j,\, k/Q_j,\, k) \qquad \text{... (8)}$$

for $j = 0, 1, 2, 3, ..., N_1 - 1$; $k = 0, 1, 2, 3, \ldots, N_2 - 1$

Where *A* is the unquantized DCT coefficients, *Q* is the quantization matrix and *B* is the quantized DCT coefficients. Decoding to display the image consists of doing all the above in reverse. Notice the slight differences between original image shown in Fig. 11. and the decompressed image in Fig. 12. at the bottom-left corner. This difference results in an error values with an average absolute error of about 5 values per pixels. The Fig. 9 and Fig. 10 shows the image after applying DCT and IDCT.

VI. RESULTS AND DISCUSSION

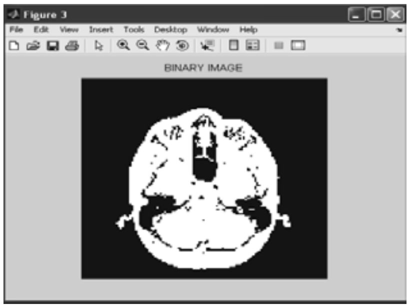

Fig. 4. Binary Image

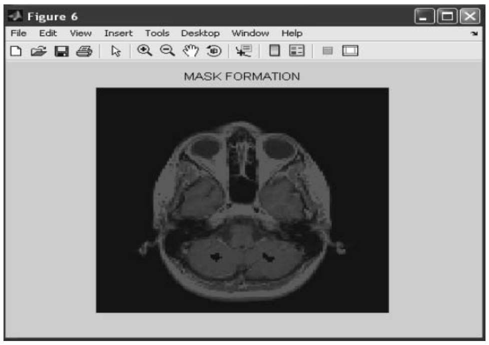

Fig. 5. Noise free Image

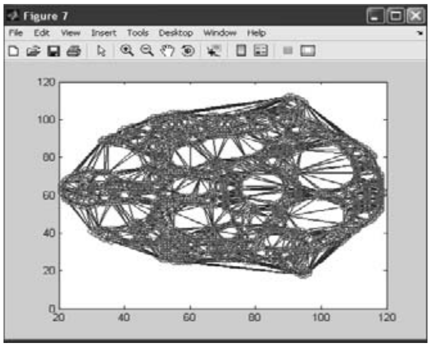

Fig. 6. Adaptive content based Mesh

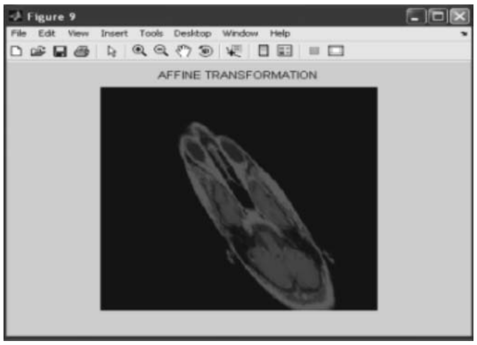

Fig. 7. Motion Compensated image

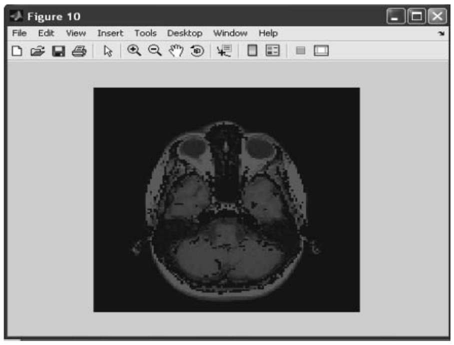

Fig. 8. Image residue

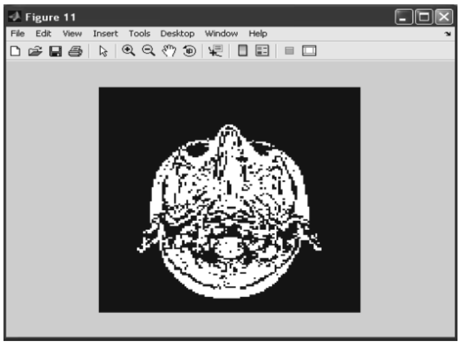

Fig. 9. After applying DCT

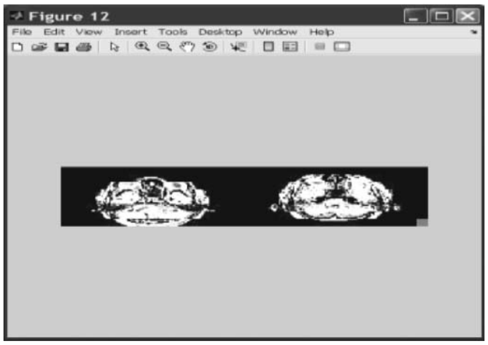

Fig. 10. After applying IDCT

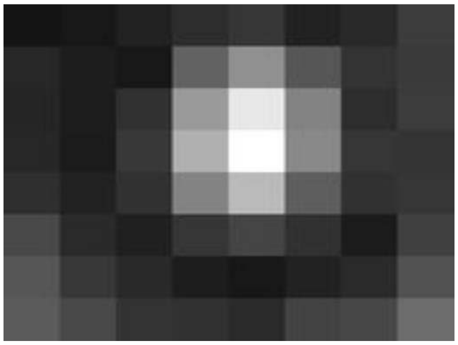

Fig. 11. Original Image

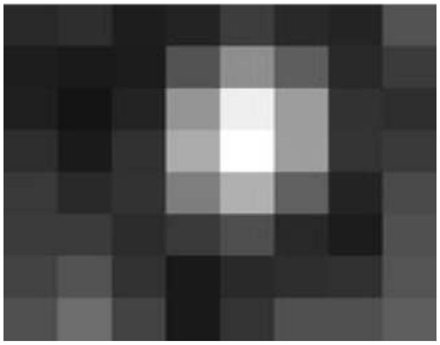

Fig. 12. Decompressed image

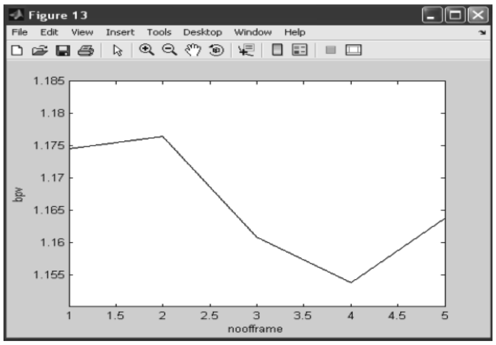

Fig. 13. A plot of no of frames Vs bpv

VII. CONCLUSION

As a part of solving the problem of storage and transmission time of the image, this paper has focused on the most optimum techniques in every aspect to compress the image. The adaptive mesh generation for the MR image sequences is proposed to avoid the multiple motions within each element of the image. To ensure the mesh connectivity in the image the BMA algorithm is used in the motion compensation as explained in (7).

A simple algorithm is also used to avoid the aperture problem in optical flow (8) estimation. Finally Context based source modeling (12)-(13),(6) is used to exploit the inter frame and intra frame correlations effectively. Further to obtain the good quality in the decompressed image JPEG compression which provides an efficient compression ratio of 4:1 is adopted. The achieved compression performance of adaptive mesh is marginally better than the uniform mesh based schemes and the 3-D wavelet based schemes (8).Fig. 13. Shows the plot for no of frames Vs bits per voxel.Compression rate achieved in our coding using JPEG is found to be 2.64 which is better compared to EZW and other wavelet schemes (8).

ACKNOWLEDGEMENT

Special acknowledgements to Dr. K.Thanushkodi for the insightful comments and suggestions.

REFERENCES

[1] Horn B.K.P. and Schunck B.G., 1981, “Determining optical flow,”*Artif. Intell.*, vol.17, pp. 185–203.

[2] Meganez G. and Thiran J.P., “Lossy to lossless object-based coding of 3-D MRI data,” *IEEE Trans. Image Process.*, vol. 11, no. 9, Sep. 2002, pp.639–647.

[3] Menegaz G. and Grewe L., “3D/2D object- based coding of head MRI data,” in *Proc.Int. Conf. Image Processing (ICIP)*, vol. 1, 2004, pp.181–184.

[4] Midtvik and Hovig I., “Reversible compression of MR images,” *IEEE Trans. Med.Imag.*, vol. 18, no. 9, Sep. 1999,pp. 795–800.

[5] Bilgin A., Zweig G. and Marcellin M.V., “Three dimensional Image compression with integer Wavelet transform,” *Applied 0ptics*, vol.39,no.11, 2000,pp.1799-1814.

[6] Nosratinia A., Mohsenian N., Orchard M.T., and Liu B., “Interframe coding of magnetic resonance images,” *IEEE Trans. Med. Imag.*, vol.15, no. 10, Oct. 1996,pp. 639– 647.

[7] Nakaya Y. and Harashima H., “Motion compensation based on spatial transformations,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, no.6, Jun. 1994,pp. 339–354.

[8] Srikanth R. and Ramakrishnan A.G., “MR image coding using content based mesh and context,” presented at the Int. Symp. Signal Processing and Applications (ISSPA-2003), Paris, France, Jul. 1–4.

[9] Taubman D., “High performance scalable image compression with EBCOT,” *IEEE Trans. Image Process.*, vol. 9, no. 5, May 1997, pp. 1158–1170.

[10] Wong S., Zaremba L., and Gooden D., “Radiological image compression—A review,” *Proc. IEEE.*, vol. 83, no. 2, Feb. 1995, pp. 194–219.

[11] Wong S., Zaremba L., and Gooden D., in the topic “Applying wavelet transforms with arithmetic coding to radiological image compression” in the year Sep/Oct 1995.

[12] Wu X. and Memon N., “Context-based, adaptive, lossless image coding,” *IEEE Trans. Commun.*, vol. 45, no. 4, Apr. 1997,pp. 437–444.

[13] Wu X.,“Lossless compression of continuous-tone images via context selection, quantization, and modeling,” *IEEE Trans. Image Process.*, vol.6, no. 5, May 1997, pp. 656– 664.

S.Bhavani: born in Coimbatore in the year 1968 received B.E degree in Electronics and Communication Engineering from V.L.B Janaki Ammal College of Engineering and Technology, Coimbatore Tamilnadu in the year 1990 and M.E in Applied Electronics from Maharaja Engineering College, Coimbatore, Tamilnadu in the year 2006. Since 1992, she was working in various disciplines in Park group of institutions, Coimbatore. Currently working as Assistant Professor in Sri Shakthi Institute of Engineering and Technology, Coimbatore. She is also a research Scholar (Part-time) in the Department of EEE at Anna University, Coimbatore Since 1997.She is also a life Member of ISTE and National Merit scholarship holder.

Dr. K. Thanushkodi: born in Theni District, TamilNadu State, India in 1948, received the BE in Electrical and Electronics Engineering from Madras University, Chennai. MSc (Engg) from Madras University, Chennai and PhD in Electrical and Electronics Engineering from Bharathiar University, Coimbatore in 1972, 1976 and 1991 respectively. His research interests lie in the area of Computer Modeling and Simulation, Computer Networking and Power System. He has published 26 technical papers in National and International Journals