

# Contextual Encoding in Uniform and Adaptive Mesh-Based Lossless Compression of MR Images

R. Srikanth\* and A. G. Ramakrishnan, *Senior Member, IEEE*

**Abstract**—We propose and evaluate a number of novel improvements to the mesh-based coding scheme for 3-D brain magnetic resonance images. This includes: 1) elimination of the clinically irrelevant background leading to meshing of only the brain part of the image; 2) content-based (adaptive) mesh generation using spatial edges and optical flow between two consecutive slices; 3) a simple solution for the aperture problem at the edges, where an accurate estimation of motion vectors is not possible; and 4) context-based entropy coding of the residues after motion compensation using affine transformations. We address only lossless coding of the images, and compare the performance of uniform and adaptive mesh-based schemes. The bit rates achieved (about 2 bits per voxel) by these schemes are comparable to those of the state-of-the-art three-dimensional (3-D) wavelet-based schemes. The mesh-based schemes have been shown to be effective for the compression of 3-D brain computed tomography data also. Adaptive mesh-based schemes perform marginally better than the uniform mesh-based methods, at the expense of increased complexity.

**Index Terms**—3-D coding, content-based mesh, context-based modeling, medical image coding, volumetric image compression.

## I. INTRODUCTION

RECENT advances in medical imaging technology have led to ubiquitous use of medical images for diagnosis. As a result, 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. With the recent developments in tele-medicine, there is also a need for reduction in transmission time. Hence, compression of medical images plays an important role for efficient storage and transmission. There are a large number of lossy schemes available for image and video compression, where the exact reconstruction of the images is not possible. In medical imaging, lossy compression schemes are not generally used prior to diagnosis so as to avoid the possible loss of useful clinical information which may influence diagnosis [1]. Hence, there is a need for efficient lossless compression schemes for medical data before diagnosis. However,

lossy compression schemes can be applied after diagnosis. Regions of interest can be compressed without any loss and the rest of the regions can be compressed with lossy schemes. Such hybrid schemes can give better compression compared to lossless schemes. In this paper, we address only the problem of lossless compression. In principle, this scheme can be extended to a hybrid scheme, as specified above.

Three-dimensional medical data are image sequences representing a 3-D organ. These images are correlated both within and across the image slices. Therefore, a 3-D compression scheme should exploit the above correlation. Some of the existing 3-D compression schemes for medical images can be found in [2]–[12]. 3-D progressive transmission schemes with lossless compression capability are reported in [3]–[8] and [10]. The schemes of [2] and [11] are based on motion compensation. In [11], the deformation across the images is modeled as motion and conventional motion compensation algorithms which assume only translation motion are employed to exploit the 3-D correlation. Although there is no motion between two successive slices, we interchangeably use the terms “motion” and “deformation.” This scheme did not give improvement over the schemes which exploit only 2-D correlation. In [2], it was observed that the simple translation motion is inadequate to model the complex deformation across the frames. A UMWC based on spatial transformation is proposed in [2] to model the deformation. The scheme uses the method proposed in [21]. Spatial transformations model rotation, translation, and scaling of image objects. However, uniform mesh elements may not adequately model the deformation because a single mesh element may contain multiple motions. Further, they use a wavelet-based lossy scheme to compress the residue after motion compensation which is not suitable for lossless compression.

We extend the scheme of [2] to lossless coding of medical images. We also propose a nonuniform mesh-based interframe coding scheme for MR sequences. The mesh is generated based upon the content of the image which ensures that multiple motions are avoided within each element. We use optical flow between two consecutive frames for mesh generation. We also propose an algorithm to avoid the classical aperture problem [19] that exists in regions of the image that have strongly oriented intensity gradients (edges) in both uniform and adaptive mesh-based schemes. Recent lossless still image compression schemes such as [3]–[7], [12]–[17] successfully employ context-based entropy coding methods. By choosing proper contexts, one can model the sources efficiently and reduce the entropy of the symbols to be coded. We employ

Manuscript received December 1, 2004; revised June 10, 2005. The Associate Editor responsible for coordinating the review of this paper and recommending its publication was M. Unser. Asterisk indicates corresponding author.

\*R. Srikanth was with the Department of Electrical Engineering, Indian Institute of Science, Bangalore 560012, India. She is now with the Department of Radiology, MRI Building, Wake Forest Medical Baptist Center Blvd., Winston-Salem, NC 27157 USA (e-mail: srikanthryali@yahoo.co.uk).

A. G. Ramakrishnan is with the Department of Electrical Engineering, Indian Institute of Science, Bangalore 560012, India (e-mail: ramkiag@ee.iisc.ernet.in).

Digital Object Identifier 10.1109/TMI.2005.853638

context-based entropy coding of the residues obtained after motion compensation.

We propose an efficient 3-D compression scheme based either on uniform or content-based nonuniform meshes, spatial transformation for motion compensation and context-based entropy coding of the residue. A part of this work is reported in [12]. We interchangeably use the terms “content-based,” “adaptive,” and “nonuniform” for adaptive mesh.

The emphasis of this paper is on efficient lossless compression. Accordingly, the only performance index we use for evaluating our schemes, as well as comparing them with existing schemes, is the lossless bit rate in bits per voxel. We do not address in detail the issues related to computational complexity of our schemes. We also have not addressed some desirable functionalities such as region of interest coding, progressive transmission, scalability and 2-D decoding from 3-D encoding. For example, [4] and [5] provide 2-D decoding from 3-D encoding as well as random access to any object of 2-D image at any desired quality.

## II. PREPROCESSING

A typical MR image consists of two parts:

- 1) air part (background);
- 2) flesh part (foreground).

The flesh part contains the useful clinical information to be compressed without any loss. The air part does not contain any clinical information; it is only noise and consumes unnecessary bits that impair the performance of a compression scheme. In [14], a scheme is proposed which uses two source models, one for background and the other for foreground. An improvement in performance is reported. However, there is no need to code the air part. This fact has been confirmed by the neuro-radiologist with whom we are collaborating. Thus, in our paper, we ignore the air part. We generate image masks in such a way that the flesh part is totally included and the pixel values in the air part are set to zero. Morphological operations can be effectively used to generate image masks, which contain a value of “1” in the foreground and a value of “0” in the background. The original image is then multiplied with this binary mask to obtain a “background noise free” image while keeping the information in the foreground intact.

The image independent algorithm for generating the mask is given below.

- 1) Binarize the image with a suitable threshold obtained from the histogram of the image.
- 2) Holes may be formed in the binary mask in the region corresponding to the foreground. Close these holes using a binary morphological “closing” operation.
- 3) The background part of the binary mask may contain spurious lines. Use a binary morphological “erode” operation to remove these lines.
- 4) The erosion operation also erodes the boundary of the foreground region of the mask. To make sure that the mask spans the entire foreground region, use a binary morphological “thickening” operation to thicken the boundary of the foreground region of the mask.

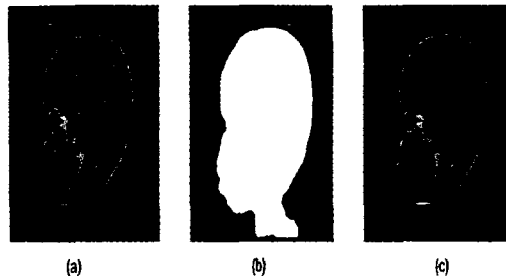


Fig. 1. Background suppression in MR images using binary morphological operations. (a) Original image (b) the generated binary mask, and (c) background suppressed image.

- 5) Now the mask contains a value of “1” corresponding to the flesh part and a value of “0” corresponding to the air part. Multiply the original image with the resulting binary mask to suppress the background.

Fig. 1 shows an MR image, its binary mask and the image obtained after multiplication with the mask. Note that this algorithm ensures that the flesh part, which is to be compressed without any loss, remains intact while the background is suppressed.

## III. CONTENT-BASED MESH DESIGN

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. In this scheme, we use triangular elements. The simplest way to divide the image is by uniform mesh elements. Mesh can also be generated based on the content of the image, which can avoid multiple deformations within a mesh element. We use both uniform and adaptive meshes in this paper and assess their relative performance.

A content-based mesh is designed such that regions with large deformations are represented by dense elements and smooth regions by few elements. Wang *et al.*, [20] proposed a scheme based on an optimization framework. In this paper, we use the scheme proposed in [22]. The basic idea is to so place the node points in such a way that mesh boundaries align with object boundaries and the density of the node points is proportional to the local deformation. The former is based on the assumption that deformation is rigid and node points are placed along the edges. The density of mesh elements is decided from the optical flow of the region. The density of elements should be high in the regions of high optical flow and vice versa. We compute the optical flow between two consecutive frames, using the method of Horn-Schunck [24]. This is a gradient-based method, which assumes conservation of intensity between two images and gives a smooth optical flow as compared to block matching algorithm (BMA) [19]. 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 information and optical flow between  $I_k$  and  $I_{k+1}$ . Let  $DFD(x, y)$  be the displaced frame difference, computed as

$$DFD(x, y) = I_k(x, y) - \hat{I}_k(x, y)$$

where

$$\hat{I}_k(x, y) = I_{k+1}(x - \delta x, y - \delta y)$$

is an estimation of  $I_k(x, y)$  based on the optical flow vector  $(\delta x, \delta y)$ .

Below, we give the procedure for mesh generation.

- 1) Label all pixels as "unmarked."
- 2) Compute the average displaced frame difference  $DFD_{avg}$  as

$$DFD_{avg} = \sum_{(x,y)} \frac{DFD^2(x,y)}{K} \quad (1)$$

where  $K$  is the number of unmarked pixels.

- 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" and it falls on a spatial edge and is sufficiently away from all the previously marked pixels.
- 5) Grow a circle around this node point until  $\sum(DFD^2(x,y))$  in this circle is greater than  $DFD_{avg}$ . 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 above algorithm 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.

#### IV. MOTION COMPENSATION USING SPATIAL TRANSFORMATION

Image compression using motion compensation methods can be defined as techniques that divide images into local regions (in our case triangles) and estimate a set of motion parameters for each region. The motion parameters are then used to estimate the current frame based on the previous frame. The difference between the actual and estimated image is then coded. The procedure that synthesizes the predicted image of the  $(k+1)$ th frame  $\hat{I}_{k+1}$  from the previous frame  $I_k(x, y)$  can be regarded as an image warping. This can be written as

$$\hat{I}_{k+1}(x, y) = I_k(f(x, y), g(x, y)) \quad (2)$$

where the geometric relationship between  $\hat{I}_{k+1}(x, y)$  and  $I_k(x, y)$  is defined by the transformation functions  $f(x, y)$  and  $g(x, y)$ . We use affine transformations for the above geometric relationships [21]. The transformation functions for the pixels in the  $i$ th triangular element are

$$\begin{aligned} f(x, y) &= a_{i1}x + a_{i2}y + a_{i3} \\ g(x, y) &= a_{i4}x + a_{i5}y + a_{i6} \end{aligned}$$

where  $a_{i1}$ – $a_{i6}$  are the deformation parameters of that element. These parameters are computed from the three node point correspondences of the  $i$ th element in  $(k+1)$ th and  $k$ th frames. These correspondences can be obtained either from the already computed optical flow or with a simple block matching algorithm. Motion vectors from the optical flow do not ensure mesh connectivity and therefore post-processing of these motion vectors

is necessary. Instead, we use the BMA algorithm which ensures mesh connectivity.

We take a  $16 \times 16$  block with the node as the center. We assume that the maximum displacement in any direction of each node is not more than 5 pixels. We move the  $16 \times 16$  block upon the next frame up to a translation of  $5 \times 5$  and take the position with the minimum mean square difference as the corresponding node point. This procedure is repeated for all nodes and the mesh elements are deformed accordingly. Let  $(u_j, v_j)$ , in the  $(k+1)$ th frame be the corresponding nodes of  $(x_j, y_j)$  in the  $k$ th frame,  $j = 1 \dots 3$ . The affine transformation can then be written as

$$\begin{pmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} a_{i1} & a_{i2} & a_{i3} \\ a_{i4} & a_{i5} & a_{i6} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{pmatrix} \quad (3)$$

This system of linear equations is solved for the parameters  $a_{i,j}$  for each element  $i$ . The motion vectors are sent as side information to the decoder. We raster scan the pixels in  $(k+1)$ th frame and find the appropriate coordinates  $(x, y)$  in the previous frame using the corresponding mesh elements affine transformation. These coordinates may not correspond to a grid pixel. In this case, we use bilinear interpolation to estimate the intensity

$$\begin{aligned} \hat{I}_{k+1}(u, v) &= \{(1 - \alpha_1) \\ &\quad \times [(1 - \beta_1)I_k(X, Y, 1) + \beta_1 I_k(X + 1, Y, 1)] \\ &\quad + \alpha_1[(1 - \beta_1)I_k(X, Y + 1, 1) \\ &\quad + \beta_1 I_k(X + 1, Y + 1, 1)]\} \end{aligned}$$

where  $(X, Y)$  is the integral part and  $(\alpha_1, \beta_1)$  is the fractional part of  $(x, y)$ . We use the rounding operator  $[\cdot]$  so that the predicted values are integers. The residue  $(r(u, v))$  as calculated below is entropy coded without any loss

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

The pixel values in smooth regions can be predicted from the previous frame using the above transformation. However, the motion vectors at intensity edges cannot be calculated accurately due to the aperture problem. To overcome this problem, we estimate edge pixels using the causal neighborhood information in the same frame as follows:

IF  $(I_{k+1}(u, v)$  is on the edge)

$$\hat{I}_{k+1}(u, v) = I_{k+1}(u - 1, v)$$

if pixel is on the horizontal edge

$$\hat{I}_{k+1}(u, v) = I_{k+1}(u, v - 1)$$

if pixel is on the vertical edge.

Here, the horizontal and vertical edges are estimated using (5), given in Section V. The thresholds for determining edge pixels are taken from [13].

#### V. CONTEXT-BASED ENTROPY CODING

The above estimation algorithm effectively exploits the linear correlation of neighborhood pixels. The residue can be compressed by entropy coders like Huffman or Arithmetic coders.

Recently, context-based entropy coders [18] have gained prominence for 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 [13] and [16], an efficient context-selection method for lossless compression of still images is given. The assumption in this method is that the errors are similar under the same context. We modify this scheme to choose 3-D contexts for entropy coding of source symbols in both the mesh-based schemes. We define two different contexts, namely, an energy context and a texture context. We then classify the error into one of the compound contexts  $(\alpha, \beta)$ , where  $\alpha$  is the energy context and  $\beta$  is the texture context. The average error for each compound context is kept track of, which is added as a bias to the predicted voxel. The resultant error is entropy coded using the energy context.

### A. Energy Contexts

The prediction error depends upon the smoothness of the image around the predicted pixel and the displacement of the pixel. We estimate the error energy as

$$\Delta = ad_h + bd_v + cm_v + d|e_w|$$

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 as follows:

$$\begin{aligned} d_h &= |I_{k+1}(x, y-1) - I_{k+1}(x, y-2)| \\ &\quad + |I_{k+1}(x-1, y) - I_{k+1}(x-1, y-1)| \\ &\quad + |I_{k+1}(x-1, y) - I_{k+1}(x-1, y+1)| \\ d_v &= |I_{k+1}(x-1, y) - I_{k+1}(x-2, y)| \\ &\quad + |I_{k+1}(x, y-1) - I_{k+1}(x-1, y-1)| \\ &\quad + |I_{k+1}(x-1, y+1) - I_{k+1}(x-2, y+1)|. \end{aligned} \quad (5)$$

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  $\Delta$  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  $\Delta$  into eight levels using the Lloyd-Max quantizer [25]. By experimenting with other slices in the two data sets, we found that the values of  $a$ ,  $b$ ,  $c$  and  $d$  vary very little; also, these variations do not affect the quantization levels of  $\Delta$  and, hence, the coding performance. Therefore, we use the above values for both the data sets.

### B. Texture Contexts

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 [16]. Let  $Q$  be a vector formed as

$$Q = \begin{bmatrix} I_{k+1}(x, y-1)I_{k+1}(x-1, y), \\ I_{k+1}(x-1, y-1)I_{k+1}(x-1, y+1), \\ I_k(x+1, y-1)I_k(x, y+1), \\ I_k(x+1, y)I_k(x+1, y+1) \end{bmatrix}.$$

We quantize  $Q$  into an 8-ary binary number  $C$ , by thresholding as follows, where  $p$  is any neighbor:

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

### C. Product Contexts

A compound context is formed by combining the texture context and four levels of the energy context. This can be viewed as a product quantization of two independent image features. We accumulate the prediction error and count the number of occurrences of each context. Adding the mean error corresponding to the current context as a bias to the prediction, the error reduces. In order to repeat this at the decoder, we compute the mean up to the previous error. This is a feedback mechanism with one time unit delay. Let  $\tilde{I}_{k+1}(x, y)$  be the corrected prediction, given by

$$\begin{aligned} \tilde{I}_{k+1}(x, y) &= \hat{I}_{k+1}(x, y) + [b_e] \\ r_c(x, y) &= I_{k+1}(x, y) - \tilde{I}_{k+1}(x, y) \end{aligned}$$

where  $b_e = S_c(\alpha, \beta)/N(\alpha, \beta)$ ,  $N(\alpha, \beta)$  being the number of occurrences,  $S_c(\alpha, \beta)$ , the accumulated error of the compound context  $(\alpha, \beta)$ , and  $r_c(x, y)$ , the error after the improved prediction. We then update the sum error and count of the context.

In addition to improving the estimated value of  $I_{k+1}(x, y)$ , we predict the sign of the residue by using the estimated mean of the present context. The sign is predicted as follows:

$$\begin{aligned} \text{IF } \{b_e(\alpha, \beta) < 0\} &\text{ send } -r_c(x, y) \\ \text{ELSE} &\text{ send } r_c(x, y) \end{aligned}$$

At the decoder, these operations are reversed by maintaining the same context errors and counts. The sign prediction reduces the entropy of the residue, since the uncertainty in the sign bit reduces. We classify the residue  $r_c(x, y)$  into eight energy contexts as described above and use arithmetic coding in each context to compress the residue.

## VI. RESULTS AND DISCUSSION

We apply both the mesh-based schemes on  $256 \times 256$ , 8-bit MR sequences with slice thickness of 1 mm provided by the National Institute of Mental Health and NeuroSciences (NIMHANS), Bangalore, and also on a data set used in [3], which is referred to as the MR-MRI data set. We also apply our schemes to  $256 \times 256$ , 8-bit 3-D computed tomography (CT) images of the brain provided by NIMHANS. All these images are first preprocessed using the method outlined in Section II. We also study the effect of preprocessing on the performance of each of the schemes. We compare the proposed schemes

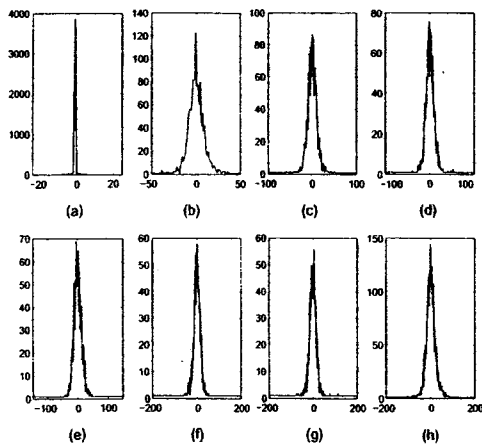


Fig. 2. Dynamic range of residue symbols for different contexts shown as histograms.

with CALIC, the state of the art 2-D lossless scheme [13]. We also compare our method with a 3-D wavelet-based coder of [3] on the MR-MRI data set. Residue symbols in each context are separately arithmetic coded. The ensuing bit rate ( $\rho$ ) in *bpv* is calculated as follows:

$$\rho = \frac{N_o + N}{256 \times 256} \quad (6)$$

where  $N_o$  is side information and  $N$  is the total number of bits required to code the symbols of all the contexts using the arithmetic coder. In the noncontext-based schemes, the residue image obtained after motion compensation is arithmetic coded without using any contexts.

We generate the content-based mesh in two ways. In the first method (Scheme A), the mesh is generated on frame  $k$  by using the optical flow between frames  $k$  and  $k + 1$ . This requires the node points to be sent as a side information in addition to the motion vectors. In the second method (Scheme B), the mesh is generated on frame  $k$  by using the optical flow between frames  $k$  and  $k - 1$ . Since the frame  $k - 1$  is available both at the encoder and the decoder, only node displacements between frames  $k$  and  $k + 1$  need to be sent. Hence, scheme B requires less side information but this is with the risk of generating a sub-optimal mesh. This risk would be higher if the optical flow between the frames  $k - 1$  and  $k + 1$  were greater.

#### A. Implementation Issues in Entropy Coding

The residue symbols in each of the eight energy contexts are entropy coded. Arithmetic coding is used since it gives flexibility in separately modeling the source and the coder. The alphabet size of the residue symbols is greater than 256 as residues take both negative and positive values. From our simulations, the dynamic range of the residues is found to be from  $-200$  to  $200$ . Fig. 2 shows the histogram of symbols in each context. It is evident that the dynamic range is much smaller within each context. For example, in context 1, most of the symbols are between  $-4$  and  $4$ . If a practical entropy coder is to be implemented, the count of each symbol has to be at least 1. Hence, a nonzero probability is to be assigned to each symbol in every context. This greatly impairs the performance of the entropy coder. To avoid

TABLE I  
PERFORMANCE (IN *bpv*) OF UNIFORM MESH-BASED SCHEME WITH CONTEXT FOR DIFFERENT NUMBER OF NODES

Number of nodes	bit-rate ( <i>bpv</i> )
64	2.39
256	2.31
1024	2.35

this problem, we use the histogram truncation technique suggested in [13] to code the symbols. In each context, we fix the alphabet size based on the statistics of the training images.

#### B. Selection of the Number of Nodes

If the distance between the nodes is small, we expect a better representation of the image. But this will increase the number of nodes and therefore the side information of the node positions to be sent to the decoder. This increase in side information may over weigh the gain obtained due to the smaller meshes. Hence, the number of nodes and the minimum distance between them need to be chosen judiciously.

To have a fair comparison between the uniform and the nonuniform mesh-based schemes, we choose the optimal number of nodes for both the schemes. We evaluate the performance of these schemes by experimenting with different number of nodes. In the case of adaptive mesh, the nodes should be selected in such a way that the entire foreground region is meshed. The minimum distance between two nodes and the maximum number of nodes are chosen as 12 and 200, respectively. The side information required for the positions of node points is about 0.1 *bpv*.

We applied uniform mesh-based scheme with context (UMWC) on NIMHANS data, using different number of nodes (namely, 64, 256, and 1024). From Table I, we see that the improvement in the coding performance of UMWC for 256 nodes over 64 nodes is about 0.08 *bpv*. For 1024 nodes, the bit rate increases by 0.04 *bpv* over that for 256 nodes. This is due to the increase in side information required to send the motion vectors. Based on these observations, we have chosen 256 nodes for uniform mesh-based schemes.

#### C. Comparison of Mesh-Based Schemes

Fig. 3 shows the bit rate for each frame of the NIMHANS data set for all the mesh-based schemes and CALIC. The increase in rate with frame number is due to the increase in the contents of the frames. Clearly, the adaptive mesh-based schemes with context perform better than the uniform mesh-based schemes and CALIC. Table II summarizes the average performance (in *bpv*) of all these schemes. The results include all the overheads. The proposed context-based schemes A and B give an average bit rates of about 2 *bpv*. Adaptive mesh-based schemes without context, AWOC and BWOC, give average bit rates of 2.55 and 2.49 *bpv*, respectively. This performance is much better than the average rate of 3.56 *bpv* achieved by the uniform mesh-based scheme without context (UMWOC). The performance of all these schemes improves by incorporating the context-based entropy coding. This significant improvement (reduction of bit rate by 1.2 *bpv*) is seen in the uniform mesh-based scheme

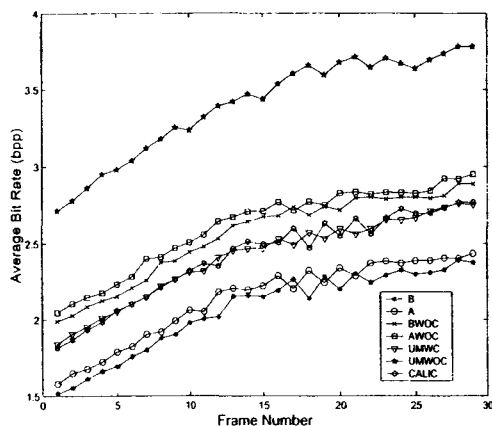


Fig. 3. Performance comparison of the various mesh-based schemes: bit rate (bpp) versus frame number. A: Proposed scheme A; B: Proposed scheme B; AWOC: Scheme A without context; BWOC: Scheme B without context; UMWOC: Uniform Mesh without context; UMWC: Uniform Mesh with context.

TABLE II

PERFORMANCE COMPARISON OF DIFFERENT SCHEMES ON BACKGROUND SUPPRESSED IMAGES. (ACRONYMS: A: PROPOSED SCHEME A; B: PROPOSED SCHEME B; AWOC: SCHEME A WITHOUT CONTEXT; BWOC: SCHEME B WITHOUT CONTEXT; UMWOC: UNIFORM MESH WITHOUT CONTEXT; UMWC: UNIFORM MESH WITH CONTEXT; UM-J2K: JPEG2000 APPLIED ON THE RESIDUE OF UMWOC; AVG. SI: AVERAGE SIDE INFORMATION)

Schemes	Avg. Bit Rate (bpp)	Avg. SI	% Pixels Coded
A	2.1	0.1	40
B	2.0	0.01	40
AWOC	2.55	0.1	40
BWOC	2.49	0.01	40
CALIC	2.41	-	100
UMWOC	3.51	0.01	100
UMWC	2.31	0.01	100
UM-J2K	2.23	0.01	100

(UMWC), because the zero-valued background is well represented by the first context.

An alternative way of effectively coding the zero symbols is with JPEG2000. We applied the JPEG2000 scheme in lossless mode after the UMWC. The average bit rate achieved is about 2.23 bpp, which is a 3.5% improvement over that of UMWC, namely 2.31 bpp. The performance obtained by schemes A and B are 2.1 and 2.0, respectively. However, it must be mentioned that JPEG2000 context models are not designed to encode residue frames.

To study the effect of the preprocessing on UMWCs, the following experiment has been performed. We applied UMWC on the MR-MRI data set both prior to and after preprocessing. Finally this scheme is applied only on the flesh part of the images (i.e., on the regions where the value of binary mask is one). The latter adds an additional overhead of about 0.01 bpp to send the coordinates of the contour of the binary mask. Table III compares the results obtained. As shown in this table, preprocessing improves the coding performance of UMWC by 0.2 bpp. Compressing only the flesh part, further improves the coding performance by 0.3 bpp. Now, this coding performance can be compared with the nonuniform mesh-based schemes, wherein only the flesh part is compressed. Scheme B on MR-MRI data set performs better than UMWC on the flesh part by only 0.2 bpp.

TABLE III

EFFECT OF BACKGROUND NOISE ON UNIFORM MESH BASED SCHEME WITH CONTEXT (UMWC). BIT RATES ARE GIVEN IN bpp FOR THE MR-MRI DATA SET. 256 AND 200 NODES ARE USED FOR THE UNIFORM AND NONUNIFORM MESH BASED SCHEMES, RESPECTIVELY. (UMWC-RAW: UMWC ON IMAGES WITHOUT PREPROCESSING; UMWC-BS: UMWC AFTER BACKGROUND SUPPRESSION; UMWC-MASK: UMWC ON FLESH PART)

UMWC-RAW	UMWC-BS	UMWC-MASK	A	B
2.67	2.48	2.16	1.98	1.89

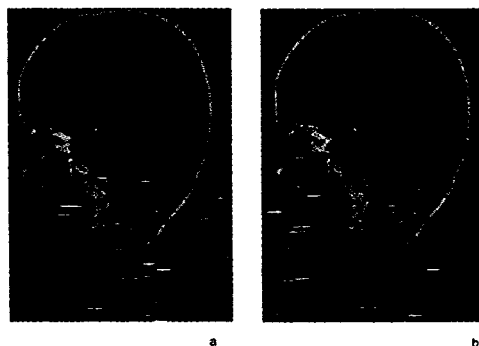


Fig. 4. Sample consecutive magnetic resonance images to be compressed: (a) slice 1; (b) slice 2. Slice 2 is to be compensated using slice 1.

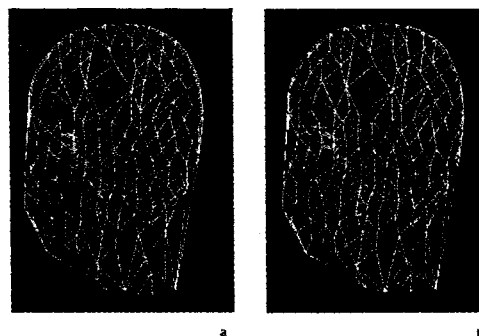


Fig. 5. An example for adaptive mesh generation using our scheme A. (a) A Content-based mesh is generated on slice 1. (b) A deformed mesh is obtained for slice 2 by finding the displacement of corresponding nodes of slice 1.

From these observations, we can infer that nonuniform mesh performs marginally better than the uniform mesh. However, adaptive mesh generation involves computationally intensive operations such as optic flow estimation, and thus, the coding gains have been obtained at the expense of increased complexity. We also see that our preprocessing improves the coding performance of UMWCs. It is to be noted that the nonuniform mesh is designed only to mesh the flesh part. The preprocessing step ensures that no nodes are selected on the air part due to some spurious lines, which we observed in our data set.

Fig. 4 shows two consecutive MR slices to be compressed. Fig. 5(a) shows nonuniform mesh on the above slices. Fig. 6 shows the residues after direct difference, motion compensation using uniform and nonuniform mesh-based motion compensation scheme with source modeling (Scheme A). Clearly, the mesh-based schemes exploit the intraframe and interframe correlations effectively.

The following reasons may account for the improved performance of adaptive mesh-based schemes.

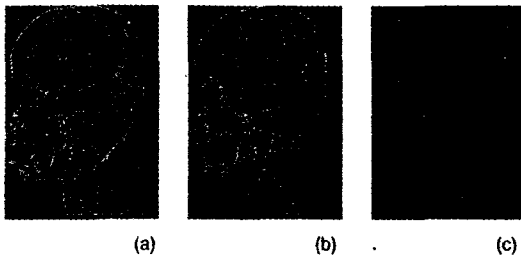


Fig. 6. Performance comparison using residues. (a) direct difference between original slices 1 and 2, (b) motion compensation using UMWC, and (c) motion compensation by our scheme A.

TABLE IV

COMPARISON OF BITRATES (IN BPV) OF SCHEMES A AND B WITH 3-D WAVELET SCHEMES ON THE MR-MRI DATA SET OF [3]. (UMWC-MASK: UNIFORM MESH WITH CONTEXT ON FLESH PART; MLZC: MULTIDIMENSIONAL LAYERED ZERO CODING; 3-D EZW: 3-D EMBEDDED ZEROTREE CODING)

A	B	UMWC-MASK	3D EZW	MLZC	JPEG2000
1.98	1.89	2.16	2.27	2.143	2.95

- 1) The uniform mesh model may be inadequate if a mesh element has multiple motions.
- 2) CALIC effectively exploits intraframe correlation but not interframe correlation.
- 3) By incorporating source models in interframe coding, both intraframe and interframe correlations are exploited.
- 4) The aperture problem in optical flow estimation is avoided by resorting to estimation of pixels on intensity gradients based on neighborhood in the same slice.
- 5) We generate nonuniform meshes in such a way that only the flesh part of the image is meshed and the air region is left out.

#### D. Comparison With Wavelet-Based Schemes

We also compared our results with the 3-D wavelet based coding scheme of [3]. For this, we applied our schemes (A, B, and UMWC) on the MR-MRI data provided by the authors of [3]. They use a 3-D embedded wavelet zerotree coder (3-D-EZW) and multidimensional layered zero coding (MLZC) for compressing MR images. Table IV summarizes the performance of these schemes. The bit rates of our A and B schemes are better than the MLZC and 3-D EZW schemes, whereas the MLZC scheme outperforms the UMWC with context applied only on the flesh part (UMWC-MASK). However, the above two sets of schemes are not directly comparable since the wavelet-based schemes code the background voxels also. An adaptive mesh is designed to code only the foreground region since the node points are selected on edges. Recently, Zixiang Xiong *et al.* [6] reported two 3-D wavelet-based coders with lossless reconstruction capability. They report a lossless bit rate of about 2 bpv on sagittal MR images which is comparable to our schemes and that of [3], although the data sets are different. Also, the wavelet-based schemes provide progressive transmission capability which is very useful in tele-radiology applications. In the present form, our schemes do not provide this functionality.

Another important aspect of 3-D compression is random access of images from the compressed 3-D volume. Currently, random access is not possible in our schemes since every frame depends on the previous frame. A simple solution to this

problem is to use intraframe coding (CALIC) on every  $n$ th slice, at the cost of bit rate. We found an increase of 0.21 bpv in scheme A and 0.27 bpv in scheme B by coding every 4th slice with CALIC.

We also applied our schemes on 3-D brain CT data. The achieved bit rates of 2.14 and 2.1 bpv, respectively, by our schemes A and B, are comparable to the values reported in [6] for brain CT data, referred to as “skull” data. However, our data set is different from the above.

#### E. Future Work

As mentioned earlier, the emphasis of this paper is on lossless coding of 3-D-MR images, which is a primary requirement of our collaborators. However, the recent 3-D compression schemes for medical images [3]–[7] provide important functionalities like region of interest coding and progressive transmission of images. The schemes in [4] and [5] also provide additional functionality of decoding 2-D images or any objects of interest from the 3-D encoded images. The current implementation of our work does not provide these important functionalities. The region of interest functionality can be provided in our scheme by coding the regions of interest without loss and the other regions with loss using appropriate quantizers. The prediction and contexts for our schemes, which are designed using lossless causal neighborhood, need to be modified accordingly. This is one of the future goals of our work. In [2], the residue after motion compensation is coded using a wavelet-based coding scheme. The images are then decoded using different resolutions of the residue. A similar scheme can also be used for the nonuniform mesh-based schemes. It will also be interesting to verify if there is any advantage in using adaptive mesh over uniform mesh for the lossy compression of volumetric images.

#### VII. CONCLUSION

Lossless uniform and adaptive mesh-based coding schemes are proposed for MR image sequences. Context-based source modeling is used to exploit the intraframe and interframe correlations effectively. In this paper, we have modified the contexts proposed in [13]. However, any other context-based modeling can be used in conjunction with both the mesh-based schemes. A simple algorithm is also given to avoid the aperture problem in optical flow estimation. The achieved lossless compression performance of adaptive mesh is marginally better than that of UMWCs and comparable with that provided by the recent 3-D wavelet-based schemes [3]. The proposed schemes also give comparable performance on 3-D CT brain images.

#### ACKNOWLEDGMENT

The authors would like to thank Prof. P. N. Jayakumar, NIMHANS, Bangalore, India, for providing the required data. They would also like to thank Dr. G. Menegaz for providing the MR-MRI data set which enabled them to compare their results with the scheme of [3]. Finally, they would like to thank the anonymous reviewers for their critical review and extremely useful suggestions, which helped them to restructure the paper and improve its presentation.

## REFERENCES

- [1] S. Wong, L. Zaremba, and D. Gooden, "Radiological image compression—A review," *Proc. IEEE*, vol. 83, no. 2, pp. 194–219, Feb. 1995.
- [2] A. Nosratinia, N. Mohsenian, M. T. Orchard, and B. Liu, "Interframe coding of magnetic resonance images," *IEEE Trans. Med. Imag.*, vol. 15, no. 10, pp. 639–647, Oct. 1996.
- [3] G. Menegaz and J.-P. Thiran, "Lossy to lossless object-based coding of 3-D MRI data," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 639–647, Sep. 2002.
- [4] —, "Three-dimensional encoding/two-dimensional decoding of medical data," *IEEE Trans. Med. Imag.*, vol. 22, no. 3, pp. 424–440, Mar. 2003.
- [5] G. Menegaz and L. Grewe, "3D/2D object-based coding of head MRI data," in *Proc. Int. Conf. Image Processing (ICIP)*, vol. 1, 2004, pp. 181–184.
- [6] Z. Xiong, X. Wu, S. Cheng, and J. Hua, "Lossy-to-Lossless compression of medical volumetric data using three-dimensional integer wavelet transforms," *IEEE Trans. Med. Imag.*, vol. 22, no. 3, pp. 459–470, Mar. 2003.
- [7] P. Schelkens, A. Munteanu, J. B. M. Galca, X. Giro-Nieto, and J. Cornelies, "Wavelet coding of volumetric image datasets," *IEEE Trans. Med. Imag.*, vol. 22, no. 3, pp. 441–458, Mar. 2003.
- [8] W.-Y. Kim, P. T. Balsara, D. T. Harper III, and J. W. Park, "Hierarchy embedded differential image for progressive transmission," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, pp. 1–13, Feb. 1995.
- [9] J. Wang and H. Huang, "Medical image compression by using three-dimensional wavelet transformation," *IEEE Trans. Med. Imag.*, vol. 15, no. 8, pp. 547–556, Aug. 1996.
- [10] P. Roos and M. A. Viergever, "Reversible 3-D decorrelation of medical images," *IEEE Trans. Med. Imag.*, vol. 12, no. 3, pp. 413–420, Sep. 1993.
- [11] —, "Reversible interframe compression of medical images: A comparison of decorrelation methods," *IEEE Trans. Med. Imag.*, vol. 10, no. 4, pp. 538–547, Dec. 1991.
- [12] R. Srikanth and A. G. Ramakrishnan, "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.
- [13] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," *IEEE Trans. Commun.*, vol. 45, no. 4, pp. 437–444, Apr. 1997.
- [14] M. Midtvik and I. Hovig, "Reversible compression of MR images," *IEEE Trans. Med. Imag.*, vol. 18, no. 9, pp. 795–800, Sep. 1999.
- [15] T. V. Rabadran and K. Chen, "The use of contextual information in the reversible compression of medical images," *IEEE Trans. Med. Imag.*, vol. 11, no. 2, pp. 185–197, Jun. 1992.
- [16] X. Wu, "Lossless compression of continuous-tone images via context selection, quantization, and modeling," *IEEE Trans. Image Process.*, vol. 6, no. 5, pp. 656–664, May 1997.
- [17] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 1158–1170, May 1997.
- [18] J. J. Rissanen, "Universal coding, information, prediction and estimation," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 629–636, 1984.
- [19] A. M. Tekalp, *Digital Video Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
- [20] Y. Wang and O. Lee, "Active mesh—A feature seeking and tracking image sequence representation scheme," *IEEE Trans. Image Process.*, vol. 3, no. 9, pp. 610–624, Sep. 1994.
- [21] Y. Nakaya and H. Harashima, "Motion compensation based on spatial transformations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, no. 6, pp. 339–354, Jun. 1994.
- [22] Y. Altunbasak and A. M. Tekalp, "Occlusion-adaptive, content-based mesh design and forward tracking," *IEEE Trans. Image Process.*, vol. 6, no. 9, pp. 1270–1280, Sep. 1997.
- [23] G. G. Langdon Jr and J. Rissanen, "Compression of black-white images with arithmetic coding," *IEEE Trans. Commun.*, vol. COM-29, no. 6, pp. 858–867, Jun. 1981.
- [24] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, pp. 185–203, 1981.
- [25] N. S. Jayant and P. Noll, *Digital Coding of Waveforms, Principles and Applications to speech and Video*. Englewood Cliffs, NJ: Prentice-Hall.