

# MR IMAGE CODING USING CONTENT-BASED MESH AND CONTEXT

*R.Srikanth and A.G.Ramakrishnan*

Department of Electrical Engineering, Indian Institute of Science, Bangalore, India  
(srikanth,ramkiag)@iisc.ernet.in

## ABSTRACT

Existing schemes for 3-D Magnetic Resonance (MR) images, such as block matching method and uniform mesh-based scheme, are inadequate to model the motion field of MR sequence because deformation within a mesh element may not all be similar. We propose a scheme consisting of (a) content-based mesh generation using optic flow between two consecutive images (b) forward motion tracking (c) motion compensation using affine transformation and (d) context-based modeling. We also propose a simple scheme to overcome aperture problem at edges where an accurate estimation of motion vectors is not possible. By using context-based modeling, motion compensation yields a better estimate of the next frame and hence a lower entropy of the residue. The obtained average compression ratio of 4.3 is better than the values of 4, achieved by CALIC, and 3, by the existing uniform mesh-based interframe coding scheme.

## 1. INTRODUCTION

In the medical image scenario, lossy compression schemes are not used due to a possible loss of useful clinical information. Operations like enhancement may result in accentuating the degradations caused by lossy compression. Hence there is a need for efficient lossless compression schemes for medical image data. Several lossless schemes based on linear prediction and interpolation [3] have been proposed. Recently, context based approach [6], has gained popularity since it can enhance the performance of the above schemes. These schemes exploit the correlation within the frame.

3-D MR images are correlated both within and across the slices. Earlier attempts to exploit the correlation in the third dimension resulted in decrease in compression performance. A close look into the MR sequences reveals that there is some deformation between two consecutive sequences. This is due to change in neuronal density from one level to another. Roos et al., [4] modeled this deformation as "motion" and employed conventional block-matching algorithm (BMA). This also resulted in reduction in performance. These schemes assume deformation (motion) due to translation only. But the deformation in MR sequences is more complex than a mere displacement. Hence, schemes

based on BMA do not adequately model the interframe deformations. Aria et al., [1] proposed a scheme based on spatial transformations that model rotation, translation and scaling to model the deformations in MR sequence. However, this model is inadequate since it uses uniform mesh elements, where in pixels within a element may have different motions.

In this paper, we propose a scheme consisting of (a) content-based mesh generation using optic flow between two consecutive images (b) forward motion tracking (c) motion compensation using affine transformation and (d) context-based modeling. We also propose a simple scheme to overcome aperture problem at edges where an accurate estimation of motion vectors is not possible. We also use context-based modeling to further improve the scheme.

## 2. CONTENT-BASED MESH DESIGN

In mesh-based schemes, the image is divided into triangular elements and the deformation of each element in the subsequent frame is modeled by a spatial transformation. In the case of uniform mesh elements, an element contains multiple deformations, and hence a spatial transformation cannot adequately model these deformations. Hence, there is a need for a content-based mesh which assigns dense elements to regions with large deformations, and few elements to smooth regions. Here, we use the scheme proposed in [7], which places 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 deformations (motion). The density of mesh elements is based on the optic flow of the region. The density is high in the regions of high optic flow and vice versa. Hence, this scheme requires the computation of optic flow between two consecutive frames.

We compute the optic flow using the method of Horn-Schunck [2]. This is a gradient-based method which assumes conservation of intensity between two images and gives a smooth optic flow as compared to BMA. Let  $I_k$  and  $I_{k+1}$  be the current and next frames respectively. Mesh is to be generated on  $I_k$  by taking its spatial information and optic flow between  $I_k$  and  $I_{k+1}$ . Let  $DFD(x, y)$  be the displaced frame difference which can be computed as

$$\begin{aligned} DFD(x, y) &= I_k(x, y) - \tilde{I}_k(x, y) \\ \tilde{I}_k(x, y) &= I_{k+1}(x - u, y - v) \end{aligned}$$

where,  $\tilde{I}_k(x, y)$  is an estimation of  $I_k(x, y)$  based on the optic flow vector  $(u, v)$ . The procedure for mesh generation is given below:

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

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

where,  $K$  is the number of unmarked pixels.

3. Find the edge map of the image using "Canny" edge operator.
4. Select a pixel as a "node" if it is "unmarked" and falls on a spatial edge and is sufficiently away from all the previously marked pixels. .
5. Grow a circle about this node point until  $\sum (DFD(x, y))^2$  in this circle is greater than  $DFD_{avg}$ . Label all pixels within the circle as "marked".
6. Go to step-2 until required number of nodes are selected, or the distance criterion is violated.
7. Given the selected nodes, apply a delaunay triangulation to obtain the required content-based mesh.

In this work, the minimum distance between two nodes is kept at 12 and a maximum of 200 nodes is chosen.

### 3. MOTION COMPENSATION

Motion compensation methods divide images into local regions and estimate for each region a set of motion parameters. The procedure that synthesizes the predicted image of the  $(k+1)$ th frame  $\hat{I}_{k+1}$  from the previous frame  $I_k$  can be regarded as an image warping. The geometric relationship between  $\hat{I}_{k+1}$  and  $I_k$  can be defined by the affine transformation [1]:

$$\begin{aligned} u &= a_{i1}x + a_{i2}y + a_{i3} \\ v &= a_{i4}x + a_{i5}y + a_{i6} \end{aligned} \quad (1)$$

where  $a_{i1}$  to  $a_{i6}$  are the six deformation parameters of the  $i$ th element.  $(u, v)$  are the coordinates in  $\hat{I}_k$  corresponding to the coordinates  $(x, y)$  in  $I_{k+1}$ . The parameters can be computed if the three node point correspondences of the  $i$ th element in  $(k + 1)$ th and  $k$  th frames are known. These correspondences can be obtained either from the computed optic flow or using simple block matching algorithm. We use the latter method since motion vectors from the optic flow do not ensure the mesh connectivity. Hence, preprocessing is needed to enforce this connectivity. Instead, we employ 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 of node is not more than 5 pixels. We move the block in the next frame within a region of  $5 \times 5$  and choose the position with minimum mean square difference as the corresponding node point in the next frame. The difference between the two positions is sent to the decoder as side information. The above procedure is repeated for all the nodes and the triangular elements are deformed accordingly.

We raster scan the pixels in  $I_{k+1}$ th frame and find the appropriate coordinate  $(u, v)$  in the previous frame using the corresponding mesh element's affine transformation. The coordinate  $(u, v)$  in  $k$ th frame may not correspond to the grid pixel. We use bilinear interpolation to estimate the intensity at these coordinates. We use the rounding operator so that the predicted values are integers so that the residue as calculated below can be entropy coded without any loss.

$$resd(x, y) = I_{k+1}(x, y) - \hat{I}_{k+1}(x, y) \quad (2)$$

The motion vectors at intensity edges cannot be calculated accurately due to the aperture problem. To overcome this problem, we estimate the pixel values at edge points using the causal neighborhood information in the same frame and pixel values in smooth regions (texture etc..) from the previous frame, where the motion estimation is accurate. The algorithm is given below:

**IF** ( $I_{k+1}(x, y)$  is on edge )  
 $\hat{I}_{k+1}(x, y) = I_{k+1}(x - 1, y)$  horizontal edge  
 $\hat{I}_{k+1}(x, y) = I_{k+1}(x, y - 1)$  vertical edge

**ELSE**

determine  $\hat{I}_{k+1}(x, y)$  from affine transformation.

The above algorithm effectively exploits the linear correlation of neighborhood pixels. The residue can be further compressed by Huffman or Arithmetic coder by assuming that the residue pixels are independent and identically distributed (memoryless). Advanced source models can be used to effectively compress the residue. We use one such source model given by Xiaolin Wu et al., [6].

### 4. SOURCE MODELING

Statistical modeling of the source being compressed plays an important role in any data compression scheme. Advances have been made [6] in building source models that can predict a symbol with a higher probability than the memoryless model and there by achieve higher compression. These models employ contexts or conditioning events to exploit intersymbol dependencies. Let  $X$  be the ensemble of sequences emitted by the source and  $Y$  be the ensemble of corresponding contexts. One can show that  $H(X) \leq H(X/Y)$ . where,  $H(X)$  is the self information and  $H(X/Y)$  is the mutual information of  $X$  given  $Y$ . Hence, by appropriately forming contexts one can reduce the entropy of the residues obtained in the last section. Here, we use the procedure given in [6].

The estimation of  $I_{k+1}(x, y)$  by the algorithm given in the previous section exploits only linear redundancy. It does not completely remove the statistical redundancy in the image sequence. The prediction error strongly correlates with the smoothness of the image around the predicted pixel  $I_{k+1}(x, y)$ . To model this correlation, formulate an error energy estimator

$$\Delta = d_h + d_v + |e_w|,$$

where  $e_w = I_{k+1}(x, y) - \hat{I}_{k+1}(x, y)$  (previous prediction error) and  $d_h$  and  $d_v$  denote estimation of horizontal and vertical edges. This energy estimator is quantized into 8 levels [6]. The residue  $resd(x, y)$  is classified into one of these bins. By conditioning the error distribution on  $\Delta$ , we can separate the prediction errors into classes of different variances. Thus, entropy coding of errors using the estimated conditional probability  $p(resd/\Delta)$  improves coding efficiency over  $p(resd)$ .

In addition to this, we can capture higher order image patterns like texture patterns by forming additional contexts  $C$ . We form these contexts  $C$  with 3 causal neighbors in the current frame and 5 neighbors in the previous frame:

$$C = [im(x, y - 1, 2) I_{k+1}(x - 1, y) \\ I_{k+1}(x - 1, y - 1, 2) I_{k+1}(x - 1, y + 1) \\ I_k(x, y) I_k(x, y + 1) \\ I_k(x + 1, y) I_k(x + 1, y + 1)].$$

We quantize  $C$  into an 8-ary binary number by using  $\hat{I}_{k+1}(x, y)$  as a threshold:

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

We form a compound context by combining the above context  $C$  with the 4 levels of energy context  $\Delta$ . Classify the error into one of the compound contexts  $(\alpha, \beta)$ , where  $\alpha$  is the energy context and  $\beta$  is the texture context. This can be viewed as product quantization of two independent image features. We accumulate the error in each context and maintain the number of occurrences of each context. We can assume that we make similar errors under the same context. Hence, by adding the mean of errors in each context as a bias to the earlier prediction, the prediction error reduces. To be able to repeat this at the decoder, we calculate the mean upto the previous error. This is a feedback mechanism with one time unit delay. Let  $\tilde{I}_{k+1}(x, y)$  denote the corrected prediction, given by

$$\begin{aligned} \tilde{I}_{k+1}(x, y) &= \hat{I}_{k+1}(x, y) + [bias] \\ resdc(x, y) &= I_{k+1}(x, y) - \tilde{I}_{k+1}(x, y) \end{aligned} \quad (3)$$

where  $bias = error(\alpha, \beta)/N(\alpha, \beta)$ , where  $N(\alpha, \beta)$  is the number of occurrences and  $error(\alpha, \beta)$  is the accumulated error of the compound context  $(\alpha, \beta)$  and  $resdc(x, y)$  is the

error after the improved prediction. Update the errors and counts of the context..

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

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

At the decoder, the reverse operation can be done by maintaining the same context errors and counts. The sign prediction helps in reducing the entropy of the residue since the uncertainty in the sign bit reduces. We classify the residue  $resdc(x, y)$  into eight energy contexts as described above and use arithmetic coding in each context to further compress the residue.

## 5. RESULTS AND DISCUSSION

We compare the proposed scheme with CALIC and uniform mesh-based interframe coding. We have used  $256 \times 256$ , 8-bit MR sequences with slice thickness of 1 mm provided by National Institute of Mental Health and NeuroSciences (NIMHANS), Bangalore. We need to send the locations of node points and motion vectors as a side information to the decoder. We generate mesh in two ways. In the first method (Scheme A), mesh is generated on frame  $k$  by using the optic flow between frames  $k$  and  $k + 1$ . This requires the node points to be sent as a side information. In the second method (Scheme B), mesh is generated on frame  $k$  by using the optic flow between frames  $k$  and  $k - 1$ . Here, only motion vectors of the nodes between frames  $k$  and  $k + 1$  need to be sent as a side information. Hence the side information to be sent is less in this case. The performance in terms of compression ratio in the Table 1 shows that there is only a marginal improvement by the second method. However, if a lossy compression were required, second method would give a better compression ratio since side information would be less.

Table-1 compares the performances (compression ratios) of the above mentioned schemes. The results include the side information for motion vectors. The compression ratios are calculated as follows:

$$compressionratio(CR) = \frac{256 \times 256 \times 8}{si + nob}$$

where  $si$  is side information and  $nob$  is the number of bits for residue after arithmetic coding.

Figure 1 shows the original two consecutive MR images. Figure 2 shows nonuniform mesh on sframes 1 and 2. Figure 3 shows the residues after direct difference, motion compensation with spatial transformation and nonuniform mesh-based motion compensation scheme with source modeling. Clearly, the modified scheme exploits the intra

and inter frame correlation more effectively and reduces the entropy of the residues.

The following reasons may account for the superior performance of the proposed scheme: (1) The uniform mesh model is inadequate since each element may have multiple motions. (2) CALIC effectively exploits intraframe correlation but not interframe correlation. (3) By incorporating source models in interframe coding, complex correlation are exploited in addition to linear correlation. (4) The aperture problem in optic flow estimation is avoided by estimating pixels on intensity gradients based on neighborhood of the same plane. (5) We generate non uniform mesh in such a way that only the object in the image is meshed and the air region is left out. This straight away improves the performance of the scheme. This kind of mesh coding can be considered as "object based coding" employed in MPEG-4 and this is achieved without any additional shape information.

## 6. CONCLUSION

The proposed scheme obtains an improved performance compared to the existing interframe coding schemes for 3-D MR images. This can also be used for lossy compression schemes. Since residue contains very little information, it can be quantized coarsely without degrading the quality of the image there by achieving high compression. The existing uniform mesh based scheme can also be improved using context-based source modeling.

Table 1: Compression ratios. (Note: "frames 1, 2" means that frame 2 is compensated based on frame 1.)

frames	calic	uniform	non uniform	Scheme A	Scheme B
1, 2	5.71	3.97	-	-	-
2, 3	5.34	3.82	4.97	5.54	5.61
15, 16	3.62	2.79	3.35	3.74	3.8
16, 17	3.62	2.72	3.32	3.73	3.78
28, 29	3.28	2.54	3.13	3.44	3.49
29, 30	3.25	2.47	3.03	3.34	3.39

## 7. REFERENCES

- [1] Aria Nosra, Nader Mohsenian, Michael T.Orchard and Bede Liu "Interframe Coding of Magnetic Resonance Images", *IEEE Trans. on Medical Imaging*, vol.15, no.5, 639-647, October, 1996.
- [2] B.K.P.Horn and B.G.Schunck, "Determining optical flow", *Artificial Intelli.*, vol.17, pp 185-203, 1981.
- [3] P.Roos and M.A.Viegever, "Reversible 3-D decorrelation of medical images", *IEEE Trans. on Medical Imaging*, vol.12, no.3, 413-420, 1993.
- [4] P.Roos and M.A.Viegever, "Reversible Interframe Compression of Medical Images: A Comparison of Decorrelation Methods", *IEEE Trans. on Medical Imaging*, vol.10, no.4, 538-547, December, 1991.

- [5] Stephen Wong, Loren Zaremba, David Gooden, "Radiologic Image Compression - A Review", *Proc. of the IEEE*, vol.83, no.2, pp. 194-219, February 1995.
- [6] Xiaolin Wu and Nasir Memon, "Context-Based Adaptive Lossless Image Coding", *IEEE Trans. on Commn.*, vol.45, no.4, pp 437-444, April, 1997.
- [7] Yucel Altunbasak and A.Murat Tekalp, "Occlusion-Adaptive Content-Based Mesh Design and Forward Tracking", *IEEE Trans. on Image Processing*, vol.6, no.9, pp 1270-1280, September, 1997.

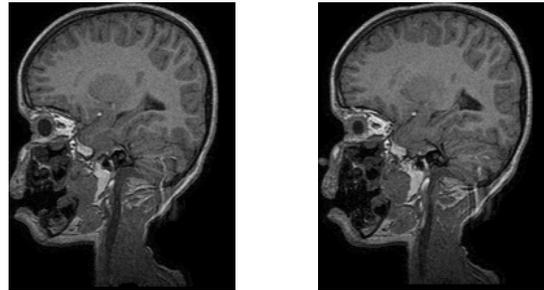


Figure 1: An example, <sup>a</sup> frame 2 to be compensated using frame 1. (a) frame 1 (b) frame 2

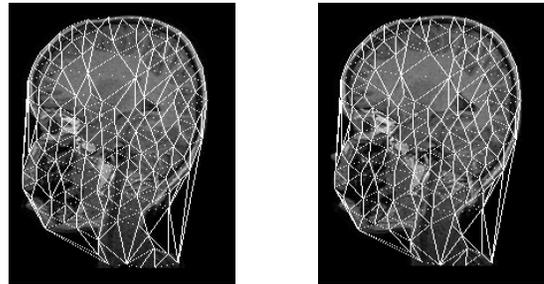


Figure 2: Content-based <sup>a</sup> mesh on (a) frame 1 (b) frame 2

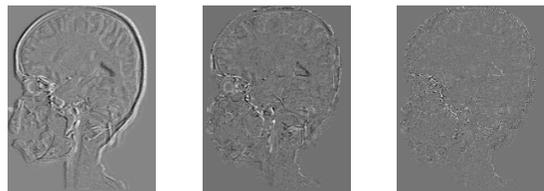


Figure 3: Residues <sup>a</sup> obtained by (a) direct difference (b) <sup>b</sup> motion compensation using spatial transformation and (c) <sup>c</sup> motion compensation by spatial transformation and Source modeling