

# Adaptive Entropy Constrained Transform Coding of Magnetic Resonance Image Sequences

Nader Mohsenian \*, Aria Nosratinia<sup>†</sup>, Bede Liu<sup>‡</sup>, and Michael T. Orchard<sup>†</sup>

\* IBM Microelectronics, Advanced Digital Video Laboratories, Endicott, NY 13760

<sup>†</sup> Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana-Champaign, Urbana, IL 61801

<sup>‡</sup> Princeton University, Department of Electrical Engineering, Princeton, NJ 08544

## Abstract

Compression of magnetic resonance images (MRI) has proved to be more difficult than other medical imaging modalities, and attempts at utilizing inter-slice dependencies for more efficient coding have so far met with little success. On the other hand, the increasing amounts of MRI data generated every day in hospitals makes this particular data compression problem very important. In this paper, we present an adaptive, entropy constrained, transform coder that also employs a new inter-slice estimator. Previous attempts at inter-slice coding of MRI have all used a piece-wise uniform, discontinuous translational model. We propose a continuous piece-wise affine model for inter-slice dependencies, whose implementation is performed through a triangle-based matching (TBM) algorithm. The residue frames from the inter-slice estimator are coded through an entropy constrained quantizer, applied to the block discrete cosine transformed (DCT) residue frame. Realizing that the statistics of the slices change spatially, one would like to spend more bits on areas containing more information. We achieve such adaptivity through dividing the image into several “activity” regions. Each activity class will have a separate bank of entropy constrained scalar quantizers (ECSQ’s) tailored to its particular average statistics. The proposed coder was used to encode a sequence of a human heart. An improvement of almost 1 *db* over non-adaptive inter-slice and 0.3 *dB* over adaptive intra-slice coding was obtained.

## I. INTRODUCTION

With recent advancements in imaging instruments and graphic workstations, it is now possible to display human organs as 3-D structures at high resolutions. Applications sprouting from these technologies call for fast image retrieval, efficient image storage, and rapid image transmission for off-site diagnosis. The amount of raw information generated with every medical image set is very large, typically on the order of tens of mega-bytes. This makes efficient image compression algorithms an indispensable part of storage, retrieval, and transmission systems.

---

\*The authors would like to thank the Siemens Corporation for providing the MR images. This work was partially supported by a grant from Motorola Inc.

In this paper, we present a new method for inter-slice coding of MRI. This problem has been open in the sense that other attempts at inter-frame coding of MRI have met with little success [1, 2]. Although fundamental information theoretical principles tell us that utilizing inter-slice information can only increase coding efficiency, inappropriate modeling for inter-slice estimation can more than offset the additional information, and result in an inferior inter-slice rate-distortion characteristic. This and other issues pertaining to the coding of MRI will be explored in more detail in the rest of the paper.

Waveform compression methods rely on the redundancies that exist by the way of dependencies and correlations in the signal, and achieve coding gain through reducing these redundancies. There exists a fair amount of correlation between neighboring pixels of a 3-D medical image sequence. This correlation can be within each slice (frame) or between two consecutive slices (frames) of an image sequence.

Most of the attempts at reducing the dependencies in the third dimension of a 3-D data set so far involve block-based matching algorithms (BMA’s) [3]. [4] applied block matching to sequences of angiograms. In [5] a set of CT image sequence is motion compensated with BMA and then a 2-D discrete cosine transform (DCT) coder is applied to the residual signal. Some coding gain was reported with this scheme in comparison to intra-slice DCT-based quantization. Other possible approaches include 3-D DCT coding [1]. However, this technique is sensitive to inter-slice distance. Generally, the sampling in the third dimension is much sparser than that of the first two and that creates problems for the 3-D DCT approach.

For 3-D MR data, the third dimension is spatial, and hence inter-slice dependencies are different from the inter-frame dependencies present in video sequences. Conventional block-based motion compensation techniques [3] can be utilized to track motion in a video sequence fairly accurately, but they do not work well with MR data, where large amounts of deformation exist between successive slices. In MRI, the piece-wise constant motion model is inadequate.

Our proposed method consists of two parts: inter-slice estimator and residue coder. The inter-slice estimator works based on a generalized piece-wise continuous affine displacement model [6]. Such a model capitalizes on the basic premise that MRI slices are representative of the

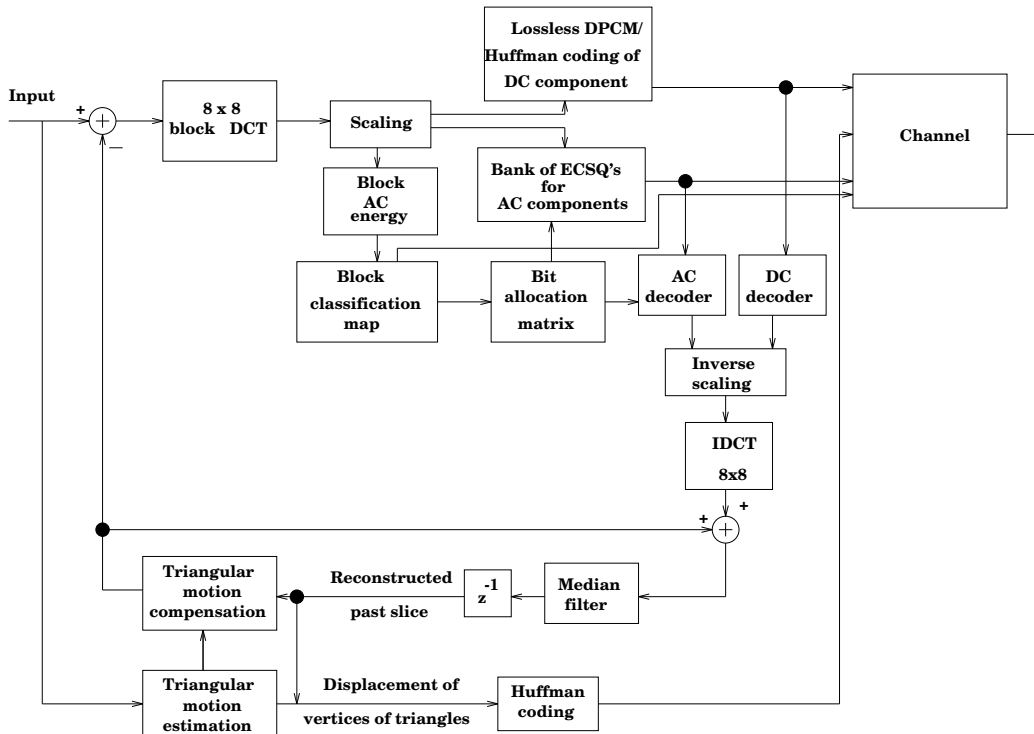


Figure 1: Block diagram of the adaptive inter-slice encoder.

slices of a solid object and hence inter-slice relationships should be continuous (unlike, e.g., video frames). The implementation of this method involves computing optimal deformations of a mesh of triangles, and is named triangle-based matching (TBM) algorithm. We also aim to minimize the effect of the large amounts of thermal noise in MRI by utilizing an in-loop filter.

The configuration of the adaptive inter-slice triangle-based (TB) encoder is shown in Fig. 1. The residual signal is formed by subtracting the motion compensated reconstructed slice from the original slice. The residue coder is an adaptive entropy constrained DCT quantizer. The DCT approximates the maximally decorrelating Karhunen-Loève transform for first order auto-regressive (AR) processes, and experience suggests that such an AR model is fairly accurate for both natural and most medical images. Unlike the K-L transform, DCT is image-independent and can be implemented efficiently. The decorrelation property of the DCT allows for efficient usage of scalar quantizers and its energy compaction property yields packing gain through thresholding and run-length coding of insignificant coefficients [7, 8].

The codebooks are designed based on the idea of entropy constrained quantization [9, 10]. Entropy constrained quantizers are obtained by a Lagrangian equalization of the marginal gain in the distortion of (frequency) bands as a function of a differential bitrate. This is in effect a bit allocation problem, and its solution is not constant over any given image. In fact, the solution is dependent on local activity levels, and we take that into account by choosing, at each location in the image, one out of several codebooks. In this way we adapt the quantization to the

local characteristics of the image.

More specifically, we classify the DCT blocks into  $L$  activity classes with an appropriate bit assignment for each class. We partition the residual image into contiguous non-overlapping blocks of size  $N_d \times N_d$ . The DCT is then applied to each block resulting in  $(N_d^2 - 1)$  AC frequency components. DC components are encoded using DPCM and Huffman coding [11]. For the AC components,  $\Psi_l(i)$  represents a bank of entropy-constrained scalar quantizers (ECSQ's) dedicated to frequency band  $i$  in blocks that belong to class  $l$ .

$$\Psi_l(i) = \left\{ \left( \mathbf{Q}_l^{(\gamma)}(i), \mathbf{C}_l^{(\gamma)}(i) \right), \gamma = 1, 2, \dots, \Gamma_{i,l} \right\}; \quad (1)$$

$$i = 1, 2, \dots, N_d^2 - 1 \quad l = 1, \dots, L$$

The quantizer/codebook pairs  $(\mathbf{Q}_l^{(\gamma)}(i), \mathbf{C}_l^{(\gamma)}(i))$  define a set of ECSQ-based encoders for frequency band  $i$  in block class  $l$ . This set consists of  $\Gamma_{i,l}$  scalar codebooks, each having  $M_{i,l}^{(\gamma)}$  reproduction levels.

The organization of the paper is as follows: Section II discusses the interslice estimation technique using deformable triangles. Section III presents adaptive entropy constrained quantization. Section IV contains simulations and comparisons with other compression schemes, and finally, discussions and conclusions are given in section V.

## II. TRIANGLE BASED MATCHING

This algorithm is a generalization of the method first proposed in [12] and later improved in [13]. The TBM algorithm partitions the current and previous slices (indexed

$k$  and  $k-1$ ) into triangular patches. The estimated version of the current frame is obtained through forming a continuous mapping of triangular regions of the past frame into present frame. The parameters of this mapping are calculated in the transmitter and transmitted to the receiver, along with the quantized estimation error.

The mapping is parameterized by the displacement vectors of the three vertices of each triangular patch. The continuity of the transform reduces the number of needed parameters in the whole frame (in addition to reflecting what we believe is the way the images are related). The displacement of any point within a given triangle can be computed in terms of the displacement of the vertices.

Let  $\vec{d}_a$ ,  $\vec{d}_b$ ,  $\vec{d}_c$  represent the displacement vectors of the vertices of an arbitrary triangle and  $\vec{d}_r$  be the displacement vector of pixel  $r$  at point  $(m_r, n_r)$  inside the triangle as in Fig. 2. Triangle  $a'b'c'$  is the transformed (predicted) version of the triangle  $abc$ . Using affine transformation  $\vec{d}_r$  is computed:

$$\vec{d}_r = (1 - p - q)\vec{d}_c + p\vec{d}_b + q\vec{d}_a . \quad (2)$$

Since affine transformation parameters  $p$  and  $q$  are independent of the individual displacements, they can be computed off-line and stored in memory to be used as a look-up table. Hence no overhead is required to transmit this information. Moreover, if we let  $(m'_r, n'_r)$  be the displaced position of pixel  $r$ , then the predicted image sample of the  $k$ th frame at point  $(m_r, n_r)$  is given by

$$\tilde{S}(m_r, n_r, k) = S(m'_r, n'_r, k - 1) \quad (3)$$

When  $m'_r$  or  $n'_r$  is not integer we use a bilinear interpolation to compute the luminance value at  $r' = (m'_r, n'_r)$ . Assume the four neighbors of  $r'$  have intensities  $S_A$ ,  $S_B$ ,  $S_C$ , and  $S_D$  as shown in Fig. 3. Then the intensity  $S_{r'}$  is

$$S_{r'} = v [u S_C + (1 - u) S_D] + (1 - v) [u S_A + (1 - u) S_B] \quad (4)$$

The displacement of the vertices is calculated in the transmitter through an iterative optimization procedure. At each iteration, all vertices of triangles are held in place except one, and that one is changed such as to minimize the mean absolute error between the two frames. To prevent wrap-overs and other undesirable effects, the allowable new positions for a moving vertex is within the inverse map of a hexagon in the present frame, which is formed by the six triangles that share the given vertex (shown as gray areas in Fig. 4). The change in the position of any one vertex affects (at most) the estimation error in an area that is bound by the same hexagon mentioned above. Therefore, computation of the cost function is performed only in this area and the error,  $\sum |S(m - \delta m, n - \delta n, k - 1) - S(m, n, k)|$ , where the sum is over all  $(m, n)$  inside the hexagon, is minimized.  $(\delta m, \delta n)$  represent the displacement vector for point  $(m, n)$ . Vertices are updated in this way in a raster scan, and re-scanned until a stopping criterion is achieved (in our case we stop after a certain number of scans).

Since the iterative procedure finds a local optimum, a good choice for the starting point can help reduce the eventual estimation error. We initialize the algorithm with a

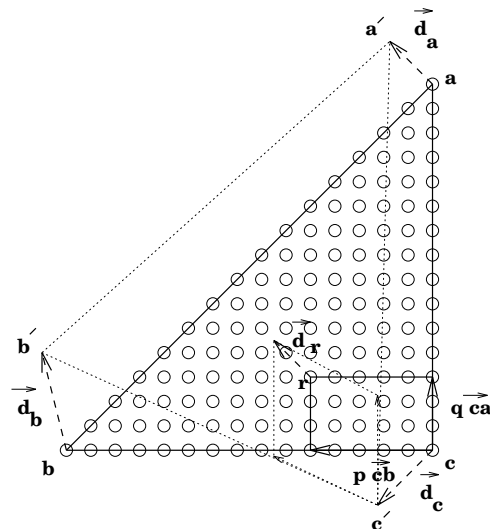


Figure 2: Displacement of pixel  $r$  in terms of displacements of vertices.

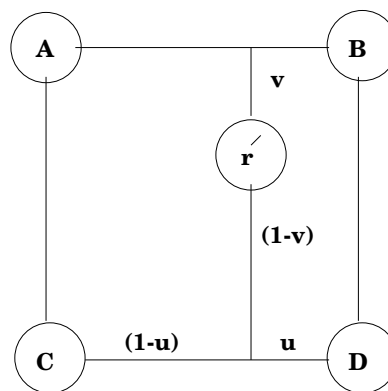


Figure 3: Bilinear interpolation for intensity at  $r'$ .

displacement that is found by a conventional block-based motion compensation method.  $16 \times 16$  blocks are centered around each vertex and the optimum minimum absolute difference position for the block indicates the starting point for the vertex. The vertices at the boundary of the image are initialized differently: their starting position is taken to be the projection of the starting position of the nearest interior vertex.

After the optimization is complete, the displaced positions of the vertices are losslessly coded with a Huffman code.

The magnetic resonance images contain a large amount of thermal noise. For practical and safety reasons, this noise cannot be reduced through cooling the reception coils (antennae). Also, there is little if any correlation between this noise and the actual images, and between the noise in two consecutive frames. The inter-frame estimator has therefore a higher error energy in the presence of the noise. In other words, the estimation error contains the addition of two noises from two consecutive frames. Moreover, the noise also reduces the quality of motion estimates. Although motion compensation and transmission of filtered

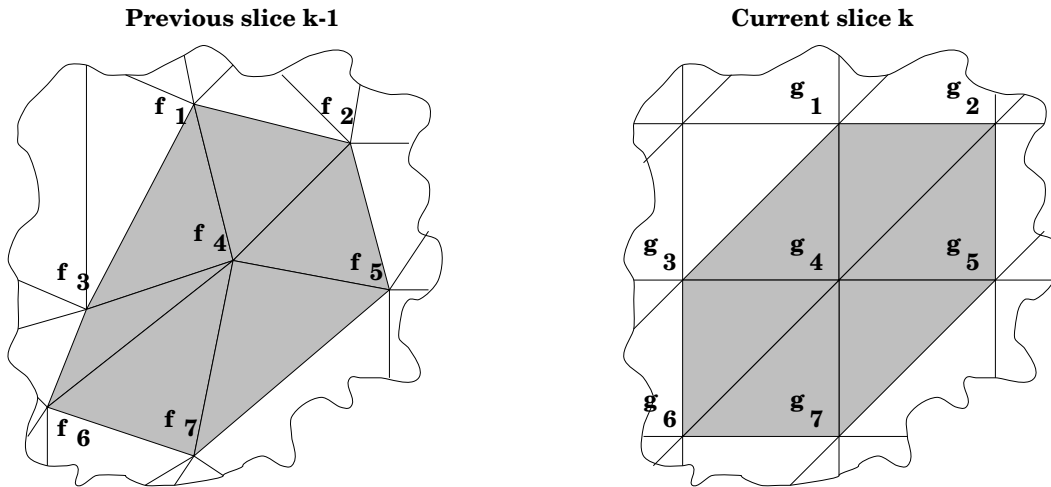


Figure 4: Regular triangular tiling of the current slice and deformation of the  $p$  previous slice.

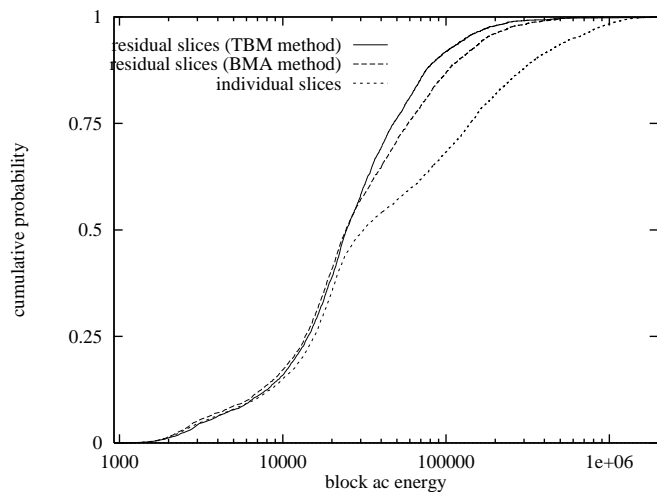


Figure 5: Cumulative probability of block AC energy.

images is generally unacceptable (because of the practical lack of information about spectral distribution of the signal and the noise, useful information may be eliminated along with the noise), it is advantageous to try and find motion correspondences that estimate a noisy present frame from a filtered past frame. This is exactly what is happening in Fig. 1. The advantage of this filtering is two-fold. In the first place, less noise in the past frame makes it easier for the iterative algorithm to find a good estimator. Secondly (as stated above), even if we know the best possible estimators in both cases, in the case where both frames are noisy the estimation error contains twice as much noise energy as the other case.

In our experiments we used a  $5 \times 5$  median filter which is characterized as follows: let  $S(m, n, k-1)$  represent the intensity of a past frame in a sequence of MR images. The filtering operation replaces  $S(m, n, k-1)$  by the median of

the pixels contained in a window around the pixel

$$S_f(m, n, k-1) = \text{median} \{S(m-\mu, n-\nu, k-1) \mid \mu, \nu \in W\} \quad (5)$$

where  $W = \{-2, -1, 0, 1, 2\}$ . Placing the filter within the motion compensation loop ensures decodability.

### III. ADAPTIVE ENTROPY CONSTRAINED TRANSFORM CODING

#### A. Transform Block Classification

The quantizer banks are designed according to a probability distribution model for the image, which is obtained from a training set. Since digital images are non-stationary in nature, a single probability distribution cannot represent all of the image very well. The coding performance can be increased over the case of a fixed quantizer by allowing the coder to adapt to the local characteristics of the image. The concept of adaptive coding is based on favoring regions of high activity in the image by allocating them more bits. Transform blocks are divided into  $L = 4$  classes according to level of image activity, measured by the total AC energy within each block. Let  $\phi(i, j)$  represent the  $ij$ -element of an  $N_d \times N_d$  image blocks, then  $\Phi(\xi, \eta)$  is the  $\xi\eta$ -element of the transformed image block and is defined by [14]

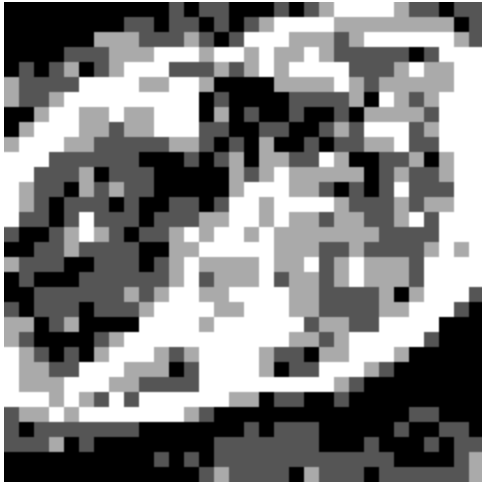
$$\Phi(\xi, \eta) = \frac{2\Theta(\xi)\Theta(\eta)}{N_d} \times \sum_{i=0}^{N_d-1} \sum_{j=0}^{N_d-1} \phi(i, j) \text{Cos} \left[ \frac{(2i+1)\xi\pi}{2N_d} \right] \text{Cos} \left[ \frac{(2j+1)\eta\pi}{2N_d} \right] \quad (6)$$

for  $\xi, \eta, i, j = 0, 1, \dots, N_d - 1$  where

$$\Theta(\omega) = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } \omega = 0 \\ 1 & \text{for } \omega = 1, \dots, N_d - 1 \\ 0 & \text{elsewhere} \end{cases}$$



(a)



(b)

Figure 6: Block classification for (a) heart slice and (b) motion compensated slice differences using the TBM algorithm.

Similarly the two-dimensional inverse DCT is defined as [14]

$$\phi(i, j) = \frac{2}{N_a} \sum_{\xi=0}^{N_a-1} \sum_{\eta=0}^{N_a-1} \Theta(\xi)\Theta(\eta)\Phi(\xi, \eta) \times \text{Cos} \left[ \frac{(2i+1)\xi\pi}{2N_a} \right] \text{Cos} \left[ \frac{(2j+1)\eta\pi}{2N_a} \right] \quad (7)$$

The activity level of a block in the image is defined as the transform domain AC energy in that block. We exclude the DC component from this activity measure, since it is coded separately and not through the quantizer banks. The total AC energy of an arbitrary block can be defined as [15]

$$E_{ac} = \sum_{\xi=0}^{N_a-1} \sum_{\eta=0}^{N_a-1} [\Phi(\xi, \eta)]^2 - [\Phi(0, 0)]^2 \quad (8)$$

Figure 5 demonstrates a typical cumulative probability distribution of AC energy of blocks. For comparison, we also show the distribution of energy for BMA residual images, and that of the original image sequence. We classify

the energy into four levels, non-uniformly separated by the .25, .5, and .75 of the cumulative probability distribution, such that each class contains  $\frac{1}{4}$ th of the blocks. This ensures that most of the image blocks use a quantizer that is most suitable to them (in a loose minimax sense). Since each classification map refers to a proper bit-allocation table, we need to send this information to the receiver as part of the overhead.

## B. ECSQ of AC Coefficients

The ECSQ design is the scalar version of the ECVQ algorithm [9, 10] and is based on joint optimization of scalar quantizers and variable-rate entropy codes. We sort the DCT blocks into 4 classes based on the thresholds computed from Fig. 5. The AC coefficients selected from each DCT block  $j$  in class  $l$  represent different frequency bands denoted by  $i$ . They are stored as 1-D sequences  $\{y_j^l(i)\}_{j=0}^{N_B/4-1}$ ;  $l = 0, \dots, L-1$  and  $i = 1, \dots, N_d^2 - 1$  such that coefficients of same band  $i$  and class  $l$  are grouped together.  $N_B$  is the total number of DCT blocks in the training sequence. For each band  $i$  in class  $l$ , ECSQ computes the optimum quantizer that minimizes the average distortion  $D_i^l$  between the sequence  $\{y_j^l(i)\}_{j=0}^{N_B/4-1}$  and the quantizer reproductions  $\hat{y}_j^l(i)$ 's

$$\min D_i^l \quad \text{subject to} \quad H_i^l \leq R_i^l \quad (9)$$

Where  $H_i^l$  is the entropy of the reproduction levels and  $R_i^l$  is the target bit-rate. The constraint in eq. (9) can be removed by introducing the Lagrangian multiplier  $\lambda_i^l$

$$\min J(\lambda_i^l) = (D_i^l + \lambda_i^l H_i^l) \quad \lambda_i^l \in [0, \infty] \quad (10)$$

The problem is now changed to minimizing the cost  $J(\lambda_i^l)$  by employing an iterative descent algorithm. For a fixed value  $\lambda_i^l$ , the solution is an optimal quantizer that solves the constrained problem (9) for some average rate of  $R_i^l$ .  $\lambda_i^l$  is swept from 0 to  $\infty$  to obtain a set of optimum quantizers. The Huffman coder is embedded into the ECSQ design algorithm, and the average Huffman code-word length is used in place of entropy  $H_i^l$ . In this manner we ensure that the actual output rate of the coder is minimized, rather than some abstract measure (entropy).

If one uses the *MSE* measure for distortion  $D_i^l$ , and let  $\ell(\hat{y}_j^l(i))$  be the number of bits generated from the Huffman routine to represent  $\hat{y}_j^l(i)$ , the Lagrangian functional to be minimized is

$$J(\lambda_i^l) = \frac{4}{N_B} \sum_{j=0}^{N_B/4-1} \{ [y_j^l(i) - \hat{y}_j^l(i)]^2 + \lambda_i^l \ell(\hat{y}_j^l(i)) \} \quad \lambda_i^l \in [0, \infty] \quad (11)$$

Starting with an initial scalar codebook of size  $M_{i,i}^{(\gamma)}$ , each sample  $y_j^l(i)$  in the training sequence is assigned to its nearest  $\hat{y}_j^l(i)$  by minimizing  $J(\lambda_i^l) = [y_j^l(i) - \hat{y}_j^l(i)]^2 + \lambda_i^l \ell(\hat{y}_j^l(i))$  where  $\ell(\hat{y}_j^l(i))$ 's are computed using the probability of the reproductions from the previous iteration and the Huffman routine. Then the new probability of

each reproduction is obtained. Finally, each reproduction is moved to the centroid of all input samples that were assigned to it and an updated codebook is generated. Each step of the iteration monotonically decreases the Lagrangian functional  $J(\lambda_i^l)$ . After convergence, the quantizer/scalar codebook pair  $(\mathbf{Q}_i^{(\gamma)}(i), \mathbf{C}_i^{(\gamma)}(i))$  solves the problem in (9) for some value  $R_i^l$  and  $\lambda_i^l$ . This scalar codebook serves as the initial codebook for the next (larger)  $\lambda_i^l$ . Once all values for  $\lambda_i^l$  are implemented,  $\Gamma_{i,l}$  encoder pairs are obtained. The ECSQ algorithm is processed for all high frequency bands, resulting in  $(N_d^2 - 1)$  different encoder banks for each block class  $l$ .

### C. Bit-allocation

The design of the individual quantizers is performed by minimizing the cost function  $J(\lambda_i^l)$  over the block DCT of a motion compensated training sequence. This procedure is carried out for all frequency bands, resulting in distortion functions  $D_i^l$ 's and fractional-rates  $R_i^l$ 's. In order to achieve good coding rates, bit-rates must be optimally distributed among the frequency bands of DCT blocks. This optimal rate distribution can be determined by minimizing distortion  $D_s$

$$D_s = \sum_{i=1}^{N_d^2-1} \sum_{l=0}^{L-1} D_i^l \quad (12)$$

subject to the condition that

$$\sum_{i=1}^{N_d^2-1} \sum_{l=0}^{L-1} R_i^l \leq R_{budget} \quad (13)$$

Where  $R_{budget}$  is the bit budget constraint. We use the near optimal pruning algorithm in [16] to minimize equation (12). For each frequency band  $i$  in class  $l$ , choose the maximum  $R_i^l$  from a set of admissible quantizers obtained in section C. This forms the initial bit-allocation table with corresponding  $D_i^l$ 's having the minimum possible values. The pruning algorithm is as follows:

1. Calculate for  $i = 1, \dots, N_d^2 - 1, l = 0, \dots, L - 1, B_i^l = \left\{ \log_2 M_{i,l}^{(1)}, \log_2 M_{i,l}^{(2)}, \dots, \log_2 M_{i,l}^{(\Gamma_{i,l})} \right\}$

$$\mathcal{S}_i(R_i^l, R_i^l - B_i^l) = \frac{D_i^l (R_i^l - B_i^l) - D_i^l (R_i^l)}{B_i^l}$$

2. For each class  $l = 0, \dots, L - 1$ , determine  $B_i^l$  for which  $\mathcal{S}_i(R_i^l, R_i^l - B_i^l)$  is minimized. Set this value to be  $B_{i^*}^l$ .
3. Determine the class  $l$  for which  $\mathcal{S}_{i^*}(R_{i^*}^l, R_{i^*}^l - B_{i^*}^l)$  is the lowest. Set  $l = l^*$  and  $R_{i^*}^{l^*} = R_{i^*}^{l^*} - B_{i^*}^{l^*}$ .
4. Calculate the total rate using eq. (13). If it equals  $R_{budget}$  stop.
5. Repeat steps 1,2,3, and 4 but do steps 1 and 2 only for class  $l^*$ .  $\mathcal{S}_i(R_i^l, R_i^l - B_i^l)$  would not have changed for other classes.

## IV. SIMULATIONS

Fig. 7 (a) depicts an original MR cross-section of the heart. The image is from a sequence of  $256 \times 256 \times 12$ -bit MR slices with 1.093 mm spatial resolution and 4 mm slice thickness. The proposed adaptive inter-slice TB coding scheme is used to encode the test sequence. As seen from Fig. 1, the previously encoded/ decoded slice is first median filtered and then covered with a regular mesh of triangles. A typical triangular tiling is depicted in Fig. 4. The size of the triangles in the TBM and blocks in the BMA results (that will be presented shortly) were chosen such that the number of motion parameters sent to the receiver are almost equal in all cases. Because the nodes at the boundary of the triangular mesh have only one degree of freedom, the TBM method turns out to have a slightly smaller set of motion parameters, but motion vectors are taken into account in computing bitrates in any case.

After the TBM algorithm is applied to predict the current slice and compute the residual image, an  $8 \times 8$  DCT block is performed on the output. In order to effectively exploit the non-stationary nature of the MR slice, we classify the DCT blocks into four activity classes with an appropriate bit assignment procedure for each class. The cumulative probability of block ac energies, shown in Fig. 5, is used to compute four thresholds for block classifications such that each region of activity is equally presented. Fig. 6 depicts the block classification for both an arbitrary slice and the motion compensated slice differences using the TBM algorithm. White represents very active regions and black is representative of low activity regions. As expected, the regions around the boundaries of the tissues have higher AC energy.

Each DCT block is encoded using the ECSQ banks and a bit-allocation table subject to a bit-budget constraint. A scalar codebook of size  $M_{i,l}^{(1)} = 256$  is designed for the full-rate quantizer ( $\lambda_i^l = 0$ ). The zero-rate quantizer corresponds to a very large  $\lambda_i^l$  and its scalar codebook has only one cell. At the decoder the entropy coded word lengths for all frequency bands are received and  $8 \times 8$  IDCT is implemented block by block to reconstruct the residual slice. The previously reconstructed slice is recalled from the memory, and is synthesized using the triangular tiling and displacement vectors of the vertices to produce a predicted version of the current slice. This predicted result is added to the residual slice to reconstruct the current slice. The heart image of Fig. 7 (a) is encoded at 1.226 bits per pixel (bpp) with  $SNR = 24.21$ , using the adaptive inter-slice TB scheme and shown in Fig. 7 (b). The encoded image is judged comparable to the original image.  $SNR$  is computed as

$$SNR = 10 \log_{10} \frac{E(\mathbf{x}^2)}{\frac{1}{N} \sum_{i=1}^N (x_i - \hat{x}_i)^2}, \quad (14)$$

where  $x_i$  and  $\hat{x}_i$  are original and reconstructed pixels and  $N = 256 \times 256$  is the size of the image.  $E(\mathbf{x}^2)$  is the expected value of random field  $\mathbf{x}$  representing the original image.

In order to demonstrate the advantage of using the TBM



(a)



(b)

Figure 7: (a) Original heart slice, (b) encoded slice at 1.226 *bpp* and  $SNR = 24.21$  *dB* using the adaptive interslice TB coding scheme.

algorithm in exploiting the inter-slice redundancies, we implemented two other coding schemes for comparison purposes. The first is an intra-slice, DCT, ECSQ-based encoder. The second scheme uses BMA and once again a DCT ECSQ. We denote the latter as adaptive inter-slice block-based (BB) coding technique. Typical block energy distributions are displayed in Fig. 5. We encoded the test sequence using the latter schemes and coding results are given in Fig. 8. As observed from Fig. 8, adaptive inter-slice BB scheme does not perform as well as other schemes at high bit-rates. The adaptive inter-slice TB scheme displays a gain of almost 0.3 *dB* over the adaptive intra-slice scheme at low rates.

Of importance is the fact that the adaptive inter-slice BB scheme does not work as efficiently as its TB counterpart. This further indicates that block-based motion estimation techniques are not suitable for removal of inter-slice dependencies in 3-D MR data. We also compare the coding results for adaptive and non-adaptive inter-slice TB

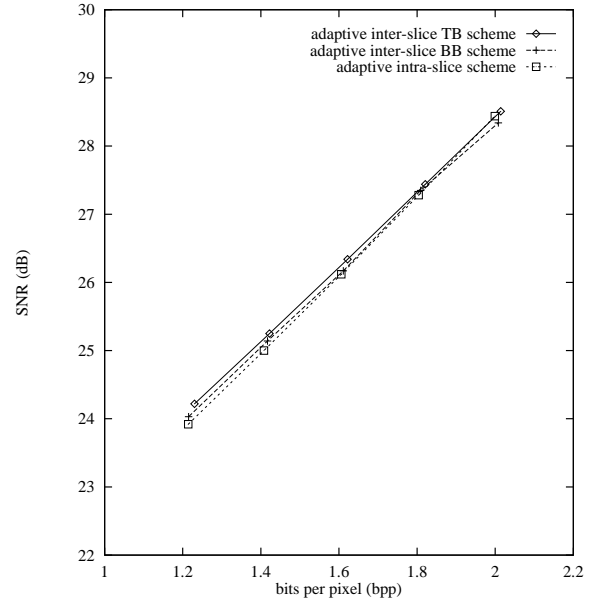


Figure 8: Average coding rate for the heart sequence for all cases.

schemes in Fig. 9. The adaptive scheme shows a gain of more than 1 *dB* over the non-adaptive scheme at low bit rates.

## V. DISCUSSION AND CONCLUSION

We have developed an adaptive coding scheme for compression of 3-D medical data where the third dimension is spatial. Image samples are de-correlated in third dimension using a triangle-based matching (TBM) algorithm. The TBM algorithm, with its continuous piece-wise affine maps, is better equipped to model the inter-slice dependencies of a 3-D object. Hence, inter-slice redundancies are efficiently exploited. Moreover, a median filter in the inter-slice estimation loop partially removes the deleterious effect of the thermal noise in MRI sequences.

Adaptivity is achieved by distributing more bits among image blocks of high activity in DCT domain. Since block classification plays an important role in performance of the adaptive encoder, a pertinent question is: how many levels of activity should one allow. With our setup, since the activity levels are coded and sent along with the message, there is a point of diminishing return where going to more activity levels (hence better matching of quantizers to local statistics) will not offset the price paid in terms of the additional bitrate. However, our experiments indicate that these activity levels are very much predictable from one slice to the next. In the case where activity levels are coded through a scheme even as simple as DPCM, the bitrate required for them could drop significantly and increasing the number of activity levels may become profitable.

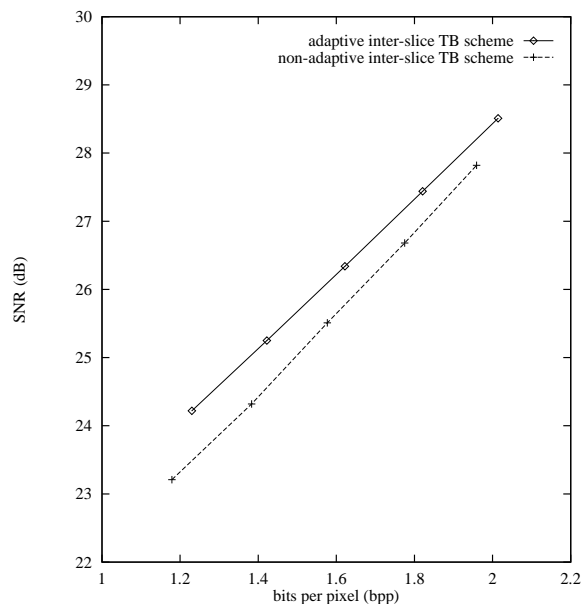


Figure 9: Comparison of adaptive and non-adaptive inter-slice TB schemes.

## References

- [1] K. K. Chan, C. C. Lau, K. S. Chuang, and C. A. Morioka, "Visualization and volumetric compression," in *Proc. SPIE, Image Capture, Formatting, and Display*, vol. 1444, pp. 250-255, 1991.
- [2] P. Roos, M. A. Viergever, M. C. A. Van Dijke, and J. H. Peters, "Reversible intraframe compression of medical images," *IEEE Transactions on Medical Imaging*, vol. MI-7, no. 4, pp. 328-336, December 1988.
- [3] H. G. Musmann, P. Pirsh, and H. J. Grallert, "Advances in picture coding," *Proc. IEEE*, vol. 73, pp. 523-548, April 1985.
- [4] P. Roos and M. A. Viergever, "Reversible interframe compression of medical images: a comparison of decorrelation methods," *IEEE Trans. Medical Imaging*, vol. MI-10, no. 4, pp. 538-547, Dec. 1991.
- [5] H. Lee, Y. Kim, A. H. Rowberg, and E. A. Riskin, "Statistical distributions of DCT coefficients and their application to an interframe compression algorithm for 3-D medical images," *IEEE Transactions on Medical Imaging*, vol. MI-12, no. 3, pp. 478-485, September 1993.
- [6] G. Wolberg, *Digital Image Warping*, IEEE Computer Society Press, Los Alamitos, California, 1990.
- [7] R.J. Clarke, *Transform Coding of Images*, Academic Press, 1985.
- [8] W. Pennebaker and J. Mitchell, *JPEG Still Image Compression Standard*, Van Nostrand Reinhold, New York, 1992.
- [9] P.A. Chou, T. Lookabough and R.M. Gray, "Entropy constrained vector quantization," *IEEE Trans. Acoust. Speech Signal Process.*, pp. 31-42, January 1989.
- [10] P. A. Chou, "Application of entropy-constrained vector quantization to waveform coding of images," *Proc. SPIE, Visual Communications and Image Processing*, vol. 1199, pp. 970-978, 1989.
- [11] R. G. Galleger, "Variations on a theme by Huffman," *IEEE Trans. Information Theory*, vol. 24, no. 6, pp. 668-674, 1987.
- [12] G. J. Sullivan and R. L. Baker, "Motion compensation for video compression using control grid interpolation," *Proc. ICASSP*, vol. 4, pp. 2713-2716, May 1991.
- [13] Y. Nakaya and H. Harashima, "Motion compensation based on spatial transformation," *IEEE Transactions on Circuit and Systems for Video Technology*, vol. 4, no. 3, pp. 339-356, June 1994.
- [14] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages and Applications*. New York, NY: Academic Press, 1990.
- [15] W-H. Chen and C. H. Smith, "Adaptive coding of monochrome and color images," *IEEE Trans. Commun.*, vol. COM-25, no. 11, pp. 1285-1292, Nov. 1977.
- [16] E. A. Riskin, "Optimal bit allocation via the generalized BFOS algorithm," *IEEE Trans. on Information Theory*, vol. IT-37, pp. 400-402, March 1991.