

Integrated Lossy, Near-lossless, and Lossless Compression of Medical Volumetric Data

Sehoon Yea^a, Sungdae Cho^b and William A. Pearlman^a

^aCenter for Image Processing Research
Electrical, Computer & Systems Engineering Dept.
Rensselaer Polytechnic Institute
110 8th St. Troy, NY 12180-3590

^bSamsung Electronics, Suwon, South Korea

ABSTRACT

We propose an integrated, wavelet based, two-stage coding scheme for lossy, near-lossless and lossless compression of medical volumetric data. The method presented determines the bit-rate while encoding for the lossy layer and without any iteration. It is in the spirit of “lossy plus residual coding” and consists of a wavelet-based lossy layer followed by an arithmetic coding of the quantized residual to guarantee a given pixel-wise maximum error bound. We focus on the selection of the optimum bit rate for the lossy coder to achieve the minimum total (lossy plus residual) bit rate in the near-lossless and the lossless cases. We propose a simple and practical method to estimate online the optimal bit rate and provide a theoretical justification for it. Experimental results show that the proposed scheme provides improved, embedded lossy, and lossless performance competitive with the best results published so far in the literature, with an added feature of near-lossless coding.

Keywords: near-lossless coding, SPIHT, lossy-to-lossless, medical image

1. INTRODUCTION

Most of today’s medical imaging techniques produce a huge amount of 3D data. Examples include Magnetic Resonance(MR), Computerized Tomography(CT), Position Emission Tomography(PET) and 3D ultrasound. Considerable research efforts have been made for their efficient compression to facilitate storage and transmission. Especially, wavelet-based coders provide a highly desirable property of progressive lossy-to-lossless compression. Lossless decoding is important from diagnostic and legal point of view, while lossy decoding allows a quick browsing through a huge volume of data.⁵ In our previous work,⁴ we proposed a wavelet-based two-stage near-lossless coding scheme. The major advantages of our proposed scheme include its simple structure and the computationally very efficient ‘on-the-fly’ wavelet-domain estimation of the optimal first-stage bit rate, which removes the need for time-consuming iterations of decoding and IDWT. In that regard, one may well expect much more significant benefits from such a scheme when it comes to the compression of huge data like 3-D medical imagery.

In this paper, we propose to use it as an alternative to the lossy-to-lossless schemes available in the 3D medical data compression literature. We straightforwardly extend the two-stage coding scheme to three-dimensional cases by adopting as the first stage coder the 3D AT-SPIHT (Asymmetric Tree Set Partitioning In Hierarchical Tree)²⁰ with the 9/7 floating-point coefficient filter. For the second stage, we use a simple arithmetic coder to encode the residual frames independently.

Experimental results show that the proposed scheme provides an improved embedded-lossy (higher and more constant PSNR, progressive up to an almost visually lossless bit-rate) and competitive lossless performances compared with the best results published so far in the literature,^{17, 20} with an added feature of near-lossless coding.

The organization of this paper is as follows. In Section 2, we review the current literature of wavelet-based 3D medical data compression and discuss several issues concerning its performance. The design of a 3D wavelet-based two-stage near-lossless coder is described in Section 3 along with discussions on the advantages of this approach. Experimental results are provided in Section 4. Section 5 concludes the paper.

Send correspondence to William A. Pearlman : E-mail: pearlw@ecse.rpi.edu, Telephone: 1 518 276 6082

2. 3D MEDICAL IMAGE COMPRESSION

Bilgin and Marcellin¹⁸ proposed a lossless medical compression scheme that is based on a 3-D integer wavelet transform and embedded zerotree wavelet (EZW) coding. They showed a better lossless compression is possible by exploiting the dependencies that exist among neighboring image slices. Even though a progressive lossy-to-lossless compression is possible, their lossy performance was compromised because their transform was not unitary. Kim and Pearlman¹⁹ proposed a 3-D lossless SPIHT based on S+P transform¹⁴ with coefficient scaling for improved lossy coding performance. Xiong *et al.*¹⁷ also proposed a general 3-D integer wavelet packet structure for bit shifting of wavelet coefficients to approximate a 3-D unitary transform along with an enhanced context-based arithmetic coding. However, there are several limitations in the above-mentioned approaches. First, as is well known, achievable lossless compression ratios are usually very limited. Second, the PSNR performance at lossy bit rates is usually compromised even with scaling of the integer wavelet coefficients and the effect of such a scaling step on lossless compression performance needs to be carefully investigated. Third, the scaling step may require additional computations (e.g. extra lifting steps) or impose an extra constraint on the level of decomposition.^{16,17} Although the evaluation of the maximum allowable distortion is still a largely open problem in the medical imaging field, a very small amount, for e.g. ± 1 or ± 2 , of *guaranteed* pixel-wise distortion is often regarded as acceptable^{5,9,11} and the benefit of allowing such ‘controlled-distortion’ is often huge. Note that traditional approaches to achieve l^∞ error control include DPCM^{11,12} and pre-quantization.⁶ In the case of DPCM, its lack of scalability makes it a less attractive option considering today’s ever-increasing demand for scalable dissemination of data. Of course, pre-quantization followed by lossless compression via an integer wavelet could provide a decent solution.⁶ However, it inherits the same problem which the integer wavelets have for lossless cases, namely compromised performance in lossy region. Also the near-lossless performance of DPCM and pre-quantization methods deteriorates rapidly as the allowable l^∞ error gets larger.^{6,9} In this light, the two-stage coding structure we will introduce in the next section has its unique place in that it really can provide an integrated lossy, near-lossless and lossless compression, all with quite competitive performances with other coders designed solely for a specific bit rate region.

3. TWO-STAGE 3D NEAR-LOSSLESS CODER

3.1. Wavelet-Based Two-Stage Near-Lossless Coder

Figure 1 shows a schematic block diagram of the two-stage near-lossless coder we used in this work.⁴ In the figure, we assume any (bi)orthogonal wavelet encoder based on successive bit-plane encoding. At the encoder, a lossy reconstruction is subtracted from the original input image to form a residual. Noting that $\|e - e_{dec}\| \leq \delta$ is equivalent to $\|I - I_{near-lossless}\| \leq \delta$, the residual e is uniformly quantized as follows to guarantee the maximum error no larger than δ :

$$\begin{aligned} e_{q-idx} &= \lfloor \frac{e + \delta}{2\delta + 1} \rfloor, & e > 0 \\ e_{q-idx} &= \lfloor \frac{e - \delta}{2\delta + 1} \rfloor, & e < 0 \end{aligned} \quad (1)$$

where $\lfloor \cdot \rfloor$ denotes the integer part of the argument. The generated quantization index e_{q-idx} is then losslessly encoded by a simple arithmetic coder without incorporating any context model.

At the decoder, we decode e_{enc} by arithmetic decoding to yield e_{q-idx} followed by a dequantizer defined as follows in order to guarantee $\|e - e_{dec}\| \leq \delta$.

$$e_{dec} = (2\delta + 1) \cdot e_{q-idx} \quad (2)$$

By adding the lossy reconstruction I_{lossy} and the dequantized residual e_{dec} , we obtain the final near-lossless reconstruction $I_{near-lossless}$ which guarantees $\|I - I_{near-lossless}\| \leq \delta$.

In a two-stage near-lossless coding scheme, one approach mentioned previously is pre-quantization, which is to quantize the source with a step size of 2δ and encode the residual losslessly.^{6,7} In such cases, efficient encoding of the given residual is the key issue, where, for example, we could build an efficient context model exploiting the

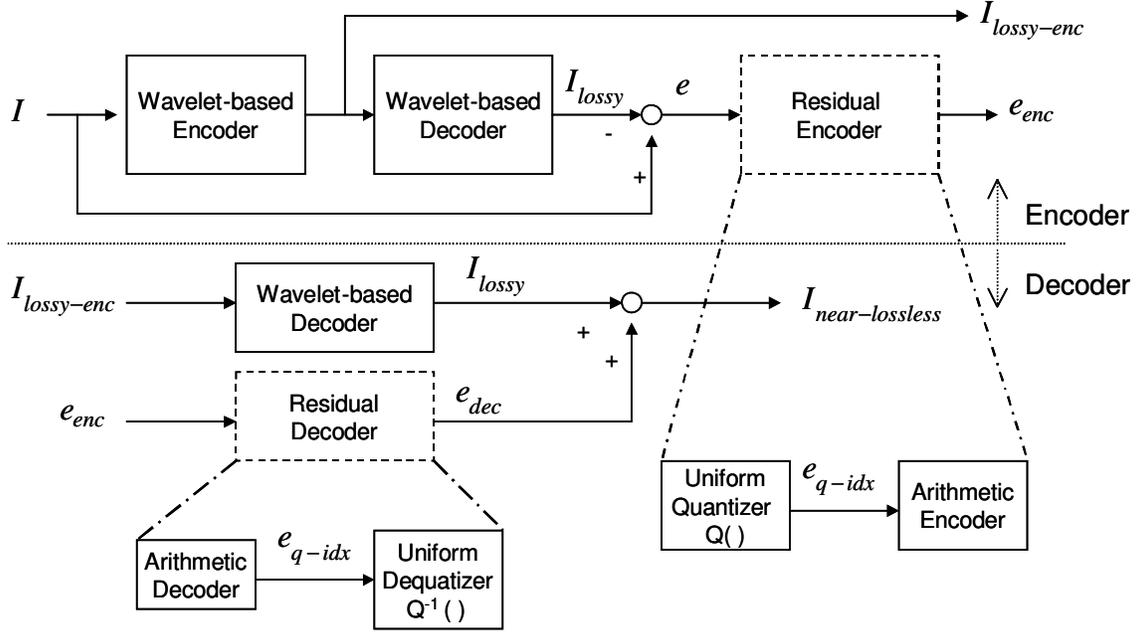


Figure 1. Two-Stage Near-Lossless Wavelet Coder

correlation between the lossy reconstruction and the residual^{6,8} or perform a search for a wavelet packet basis for the residual.⁷ However, if we want the total (lossy plus residual) bit-rate to be minimized for any given δ , then pre-quantization plus lossless coding will not achieve this goal. The approach taken here, similar to other approaches, e.g., Ansari *et al.*,⁶ is to determine the first-stage lossy layer rate that minimizes the total bitrate of the the lossy layer plus the residual. Our method is the only one that determines this optimal first stage lossy rate during encoding without exhaustive iteration.

Figure 2 shows the central idea of our method to determine the optimal first-stage rate. When the bit rate (R) for the lossy reconstruction (I_{lossy}) becomes larger than the ‘critical rate’ (R_c), the encoding residual of a source becomes white,^{1,2} and the residual and its wavelet transform converge to each other in probability distribution.⁴ Therefore, the first-order entropy $H_1(Y)$ of the wavelet transform residual and the first-order entropy of its inverse, the source residual’s entropy $H_1(X)$ are equal.³

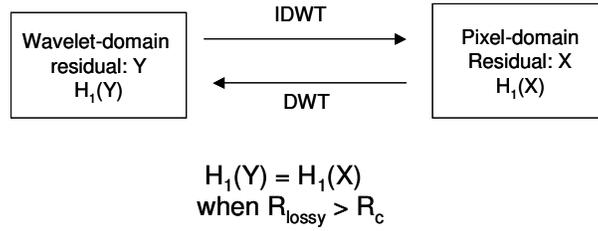


Figure 2. First-Order Entropies in Two Domains

Above this critical rate R_c , the white (uncorrelated) residual lacks the structure that a good lossy encoder (e.g., SPIHT) can take advantage of, so the coding efficiency of such a coder tends to become at most as good as and usually worse than that of first-order entropy coding. Therefore, assuming a first-order encoding of the (quantized) residual by the arithmetic coder, the total bit rate curve has a point or a small flat region of bit rates where it attains its minimum. Indeed we have confirmed experimentally the convergences of the first-order entropies $H_1(X)$ and $H_1(Y)$ of the source and wavelet residuals and their quantized ($\delta = 1$) counterparts, $H_1(\hat{X})$ and $H_1(\hat{Y})$ beyond the rates R_c and \hat{R}_c , respectively. (The quantization is that done in Equation (1)). In Figure 3,

we have plotted the residual coding rate and the total bitrate versus the lossy bitrate for the lossless ($\delta = 0$) case and the near-lossless ($\delta = 1$) case. This figure pertains to coding of slices 16 to 31 of the SKULL sequence (see Figure 4(a)), but the conclusions drawn hold true in general. In Figure 3(a), we see the convergences of the residual coding rates in both domains above a certain rate. Note that the quantized residual ($\delta = 1$) case shows faster convergence than the lossless ($\delta = 0$) case. We refer the reader to our previous paper for a proof of this circumstance.⁴

Figure 3 (b) shows the convergences of the total (lossy plus first-order residual) bit rates above the lossy rate R_c or \hat{R}_c , where the total rate reaches a minimum and starts to increase. This phenomenon forms the basis of our method of identifying the optimal lossy rate point, either R_c or \hat{R}_c for lossless and near-lossless coding, respectively. In the sequel, the term 'quantized' will refer to the case with any $\delta \geq 0$, including $\delta = 0$, corresponding to the 'unquantized' or lossless case.

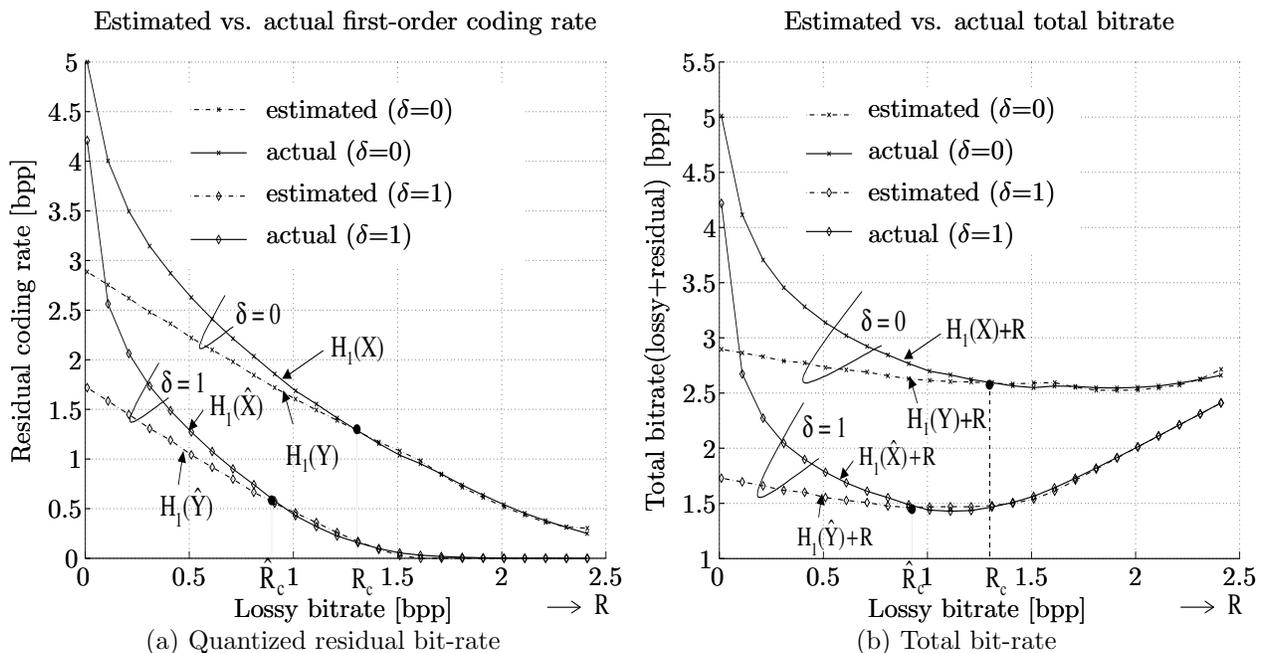


Figure 3. Estimated vs. actual bit-rates for SKULL(slices:16-31)

3.2. On the Fly Estimation of Optimal Lossy Bit Rates

The fact that the total bit rate (lossy layer bit rate R plus first-order entropy of the quantized residual $H_1(\cdot)$) versus lossy layer bit rate curves in both domains converge implies that we only need to know the first-order entropy ($H_1(\hat{Y})$) of the quantized wavelet-domain residual in order to estimate the actual total bit rate ($R + H_1(\hat{X})$). This implies we do not need to generate lossy reconstructions at various bit rates and take the inverse wavelet transform to generate the quantized pixel-domain residuals at corresponding bit rates. Potential savings in computation obtained thereby can turn out to be significant especially when dealing with 3D sequences.

Algorithm 1 describes the 'on-the-fly' estimation of the first-order entropy of the quantized wavelet domain residual ($H(\hat{Y})$) as we encode with SPIHT. The idea of the algorithm is that only two symbol probabilities (P_i 's) of the quantized wavelet-domain residual need to be updated at any particular moment of successive bit-plane encoding in order to keep track of the corresponding change in the first-order entropy ($H(\hat{Y})$), where P_i is given as the frequency count (FR_i) of the corresponding quantized wavelet-domain residual value divided by the total number (N) of the wavelet coefficients. Thus, it is readily understood that the proposed estimation algorithm requires almost negligible computational effort.

Finally, we would like to point out that, when estimating the first-order entropy $H(\hat{Y})$, we first need to round the floating point value of the wavelet domain residual to the nearest integer before applying the uniform quantization rule in Equation (1).

Algorithm 1 Iterative First-Order Residual-Entropy Estimation ‘on the fly’

Step 1: Given a maximum-error bound δ , initialize the first-order entropy of the quantized wavelet coefficients as follows:

$$H(\hat{Y}) = - \sum_i P_i \log P_i$$

, where P_i is the probability of the i -th symbol for the quantized wavelet coefficients.

Step 2: Everytime a wavelet coefficient is further encoded via updating the significance/refinement information inside the SPIHT, save the symbol index of the quantized value of that wavelet coefficient before updating as *old* and find the corresponding one after updating as *new*.

Step 3: Subtract the contribution of the above symbols from the first-order entropy estimate as follows:

$$H(\hat{Y}) \leftarrow H(\hat{Y}) - P_{old} \log P_{old} + P_{new} \log P_{new}$$

Step 4: Update the symbol probabilities of the symbols *old* and *new* as follows:

$$FR_{old} = FR_{old} - 1, FR_{new} = FR_{new} + 1$$

$$P_{old} = FR_{old}/N, P_{new} = FR_{new}/N$$

where FR_i is the frequency count of the i -th symbol and N is the total number of the wavelet coefficients.

Step 5: Add the new contribution of the updated symbol probabilities to the first-order entropy estimate as follows:

$$H(\hat{Y}) \leftarrow H(\hat{Y}) - P_{old} \log P_{old} - P_{new} \log P_{new}$$

Step 6: Repeat steps 2 through 5 everytime a wavelet coefficient is further encoded.

4. EXPERIMENTAL RESULTS

In this section, we present the experimental results on the proposed two-stage near-lossless coding with 3D medical data sets. Figure 4 depicts eight medical test-images we used for experiments. They are 256×256 images with 8-bit gray-scale pixel values. Throughout the experiments, we used the GOS (Group Of Slices) size of 16 for all the test sequences.

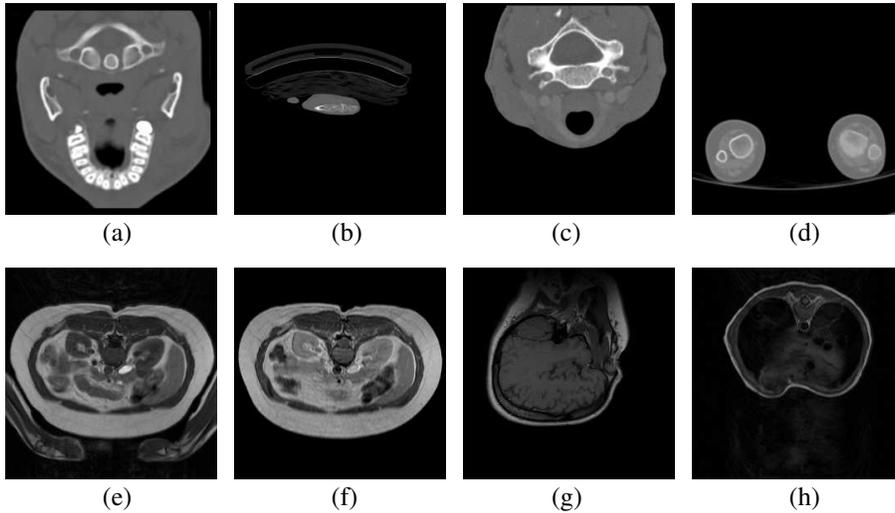


Figure 4. Volumetric medical images : First slice of each data set. (a) Skull (b) Wrist (c) Carotid (d) Aperts (e) Liver_t1 (f) Liver_t2e1 (g) Sag_head (h) Ped_chest

4.1. Compression Ratio Improvement

In this subsection, we look at the bit rates for lossless and near-lossless compression. Table 1 compares the lossless compression bit rates of the proposed scheme with those of other coders. In the table, the columns ‘3D SPIHT’ and ‘3D AT-SPIHT’ refer to the results with lossless 3-D SPIHT with S+P filter, each based on the symmetric and the asymmetric tree structure, respectively.²⁰ The columns ‘3D ESCOT’¹⁷ and ‘3D CB-EZW’¹⁸ refer to the lossless results with 3-D ESCOT and 3D CB-EZW both with S+P filter, respectively. The column under ‘2-D NL’ corresponds to the use of 2-D two-stage near-lossless coder for each slice, i.e. intra-frame encoding. The last column shows the results with the proposed 3-D AT-SPIHT-based two-stage near-lossless coder. In all the aforementioned 3-D cases, the size of GOS is 16 and the average bits per pixel (bpp) is over all GOS’s. From the table we see our coder performed the best for 2 out of 8 test sequences, with comparable bit rates to other lossless coders in all the other test sequences. Keep in mind that all the other 3-D wavelet coders do not have the capability of near-lossless coding and especially ‘3D ESCOT’ and ‘3D CB-EZW’ incorporate high-complexity context-modelling techniques to achieve extra savings in bit rates. Also note all the 3-D coders significantly outperformed the 2-D one in the table thanks to their ability to take advantage of the inter-slice correlation. Next, Table 2 shows the bit rates for various sizes of the maximum allowable error δ . Note especially that by allowing only $\delta=1$, we have a dramatic reduction in the required bit rates compared with the lossless case of $\delta=0$.

Name	3D SPIHT	3D AT-SPIHT	3D ESCOT	3D CB-EZW	2-D NL($\delta=0$)	3-D NL ($\delta=0$)
Aperts	1.0543	1.0397	0.9	1.0139	1.4239	0.9333
Carotid	1.4977	1.4973	1.4	1.4553	2.3179	1.6500
Liver.t1	2.3997	2.3690	2.2	2.4156	3.3998	2.3767
Liver.t2	1.7483	1.7440	1.6	1.7530	2.8342	1.8567
Ped.Chest	2.1045	2.0540	1.9	2.1174	3.0564	1.7700
Sag.Head	2.2400	2.2023	2.0	2.3569	2.9770	2.1367
Skull	2.1134	2.1661	2.0	2.2046	3.5260	2.4050
Wrist	1.3689	1.3550	1.2	1.3274	2.0643	1.1445

Table 1. Bit rates for lossless compression, average bitrate in bits per pixel (bpp), GOS=16

Name	$\delta=0$	1	2	3	4	5	6
Aperts	0.9333	0.3883	0.2717	0.1800	0.2117	0.1800	0.1550
Carotid	1.6500	0.8350	0.5850	0.4150	0.3450	0.2900	0.2475
Liver.t1	2.3767	1.0933	0.7167	0.5033	0.4200	0.3433	0.2867
Liver.t2	1.8567	1.0433	0.7767	0.5967	0.5233	0.4433	0.3700
Ped.Chest	1.7700	0.7000	0.4900	0.3200	0.2750	0.2125	0.1750
Sag.Head	2.1367	1.1133	0.8200	0.5833	0.4867	0.3933	0.3267
Skull	2.4050	1.3713	1.0350	0.7812	0.6725	0.5637	0.4812
Wrist	1.1445	0.5118	0.3645	0.2582	0.2218	0.1809	0.1545

Table 2. Bit rates for near-lossless compression, average bitrate in bpp

4.2. Improved Low Bit Rate PSNR Performance

In this subsection, we show the benefits of using the floating-point coefficient filter in terms of the PSNR performance in the lossy bit rate region. Since most integer wavelet filters are non-unitary, ‘scaling’ is usually done on the integer filter coefficients in order to approximate the unitary ones^{17,20} and thereby improve the low bit rate PSNR performance. Table 3 compares PSNR’s at low bit rates (0.1 bpp and 0.5 bpp) between the coders based on integer filters and the one based on the floating-point filter. The first and second rows in the table correspond to the cases where the (2+2,2) filter (with the ‘scaled coefficients’) with a four-level integer wavelet packet transform is used in 3-D SPIHT and 3-D ESCOT encoding, respectively.¹⁷ The improvement in PSNR’s of the latter over those of the former is thought to be mainly due to the more sophisticated context-modelling.

The third row shows the PSNR results with the first stage of our proposed scheme, i.e. 3-D AT-SPIHT using the 9/7 floating-point filter. Note 0.1 bpp and 0.5 bpp are typical bit rates where only the first stage (i.e. the lossy stage) of our scheme is used. From the table, even with scaling for integer filters, we can see that there still is up to a few dB advantage at low bit rates in the 9/7 filter cases over the integer filter ones. Considering the fact that the 9/7 filter and the SKULL images do not match well, the PSNR difference could be more significant in other sequences.

	0.1 bpp	0.5 bpp
3-D SPIHT (LL) ¹⁷	33.99 dB	42.89 dB
3-D ESCOT (LL) ¹⁷	34.68 dB	43.82 dB
3-D AT-SPIHT (LS)	35.27 dB	45.38 dB

Table 3. PSNR Comparison, LL=Integer Filter(Scaled Coefficients), LS=Floating-Point Filter

4.3. Constant Quality around GOS Boundaries

In a 3D sequence coding, there occurs a severe drop in PSNR at every GOS boundary. This may cause very annoying jittering artifacts in motion video playbacks or unacceptable irregularity of quality in medical imagery. The reason for this phenomenon is the limited size of the GOS due to delay or memory constraints. Xu and Xiong *et al.*²¹ proposed a lifting-based memory-constrained video coding scheme to alleviate this problem. However, it normally requires more memory than 16 frames. A. Golwelkar also implemented the ‘sliding-window’ approach to mitigate the boundary effect.²² We can see from Figure 5 that PSNR drops at the boundaries are significantly reduced in our near-lossless coding scheme. This is a natural byproduct of the near-lossless coding where local error level is strictly upper-bounded. Notice that we have an average of 0.98 dB lower PSNR with the near-lossless coding than with the pure lossy coding (using the same bit rates as the lossy plus residual rates in near-lossless case). Although guaranteeing the pixel-wise l^∞ error norm does not ensure higher or equal average PSNR’s, it does guarantee more constant PSNR values, in general. The bottom line is that we do not need any extra memory or incur extra encoding delay. Besides, the aforementioned works aimed at reducing the boundary drop will in fact act additively with our scheme since they are improving the PSNR response of the lossy part given the same bit rate.

5. CONCLUSION

We propose an integrated, wavelet based, two-stage coding scheme for lossy, near-lossless and lossless compression of medical volumetric data. The method presented *determines the bit-rate while encoding for the lossy layer and without any iteration.* It is in the spirit of “lossy plus residual coding” and consists of a wavelet-based lossy layer followed by an arithmetic coding of the quantized residual to guarantee a given pixel-wise maximum error bound.

Experimental results show that the proposed scheme provides an improved *embedded-lossy* (higher and more constant PSNR, progressive up to an almost visually lossless bit-rate) and competitive *lossless* performances compared with the best results published so far in the literature, with an added feature of *near-lossless* coding.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support of the National Geospatial-Intelligence Agency under Grant No. NMA201-00-1-2008.

REFERENCES

1. T. Berger, *Rate-Distortion Theory*, Prentice-Hall, 1971.
2. Jianping Pan and Thomas R. Fischer, “Two-Stage Vector Quantization - Lattice Vector Quantization,” *IEEE Trans. on Inform. Theory*, Vol. 41, No. 1, pp-155-163, 1995.
3. Padmanabha Rao and William A. Pearlman, “On Entropy of Pyramid Structures,” *IEEE Trans. on Inform. Theory*, Vol. 37, No. 2, pp. 407-413, March 1991.

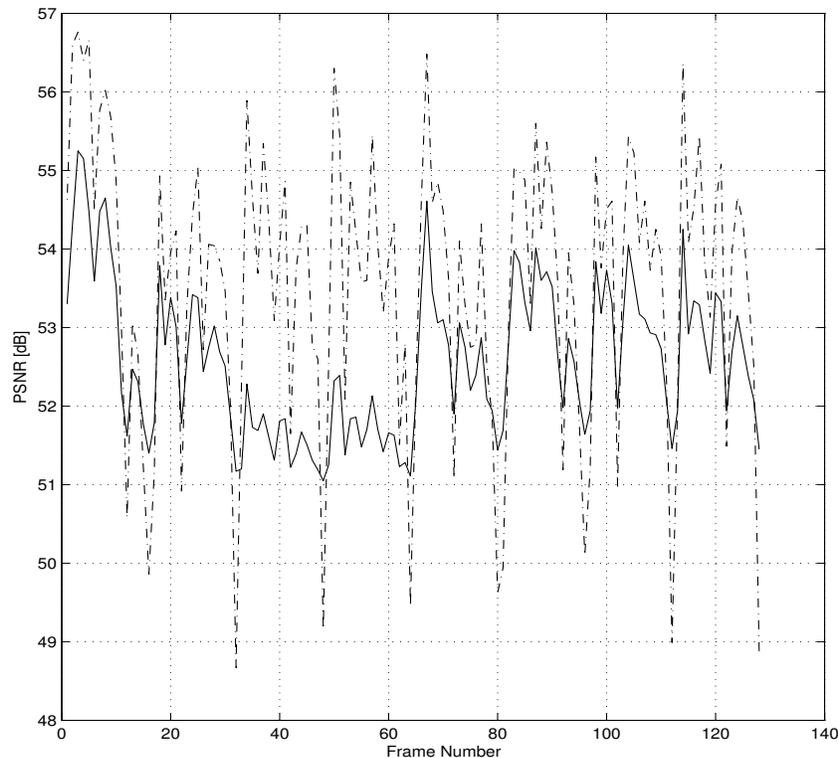


Figure 5. PSNR Comparison, Skull, GOS=16, solid: near-lossless coding with $\delta=1$ (avg:52.58 db), dash-dot: lossy only, using the same bit rates as the near-lossless (avg:53.56 dB),

4. Sehoon Yea and William A. Pearlman, "A Wavelet-Based Two-Stage Near-Lossless Coder," *Proc. of IEEE Intl. Conf. on Image Processing*, Singapore, 2004.
5. P. Cosman, R. Gray and R. Olshen, "Evaluating quality of compressed medical images: SNR, subjective rating, and diagnostic accuracy," *Proceedings of the IEEE*, Vol. 82, No. 6, pp. 919-932, June 1994.
6. R. Ansari and N. Memon and E. Ceran, "Near-lossless image compression techniques," *J. Electronic Imaging*, Vol. 7, No. 3, pp. 486-494, July 1998.
7. Detlev Marpe and Gabi Blattermann and Jens Rieke and Peter Maab, "A two-layered wavelet-based algorithm for efficient lossless and lossy image compression," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 10, No. 7, pp. 1094-1102, Oct. 2000.
8. Alexandre Krivoulets, "A method for progressive near-lossless image compression," *Proc. of IEEE Int. Conf. Image Processing*, pp. 185-188, 2003.
9. X. Wu and P. Bao, " L^∞ Constrained High-Fidelity Image Compression via Adaptive Context Modeling," *IEEE Transactions on Image Processing*, Vol. 9, No. 4, pp. 536-542, April 2000.
10. M. Weinberger, G. Seroussi and G. Sapiro, "The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS," *IEEE Transactions on Image Processing*, Vol. 9, No. 8, pp. 1309-1324, August 2000.
11. L. Ke and M. Marcellin, "Near-Lossless Image Compression: Minimum-Entropy, Constrained-Error DPCM," *IEEE Trans. on Image Processing*, Vol. 7, No. 2, pp. 225-228, Feb. 1998.
12. K. Chen and T. Ramabadran, "Near-Lossless Compression of Medical Images Through Entropy-Coded DPCM," *IEEE Trans. on Medical Imaging*, Vol. 13, No. 3, pp. 538-548, Sept. 1994.
13. A. Said and W. Pearlman, "New, Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 6, pp. 243-249, June 1996.
14. A. Said and W. Pearlman, "An Image Multiresolution Representation for Lossless and Lossy Compression," *IEEE Transactions on Image Processing*, Vol. 5, pp. 1303-1310, Sept. 1996.

15. B.J. Kim, Z. Xiong and W.A. Pearlman, "Low Bit-Rate Scalable Video Coding with 3-D SPIHT," *IEEE Transactions on CSVT*, Vol.10, , pp.1374-1387, December 2000.
16. M. Grangetto and E. Magli et al., "Efficient common-core lossless and lossy image coder based on integer wavelets," *Signal Processing* ,Vol.81, pp.403-408, 2001
17. Zixiang Xiong, Xiaolin Wu, Samuel Cheng and Jianping Hua, "Lossy-to-Lossless Compression of Medical Volumetric Data Using Three-Dimensional Integer Wavelet Transforms," *IEEE Trans. on Medical Imaging*, Vol. 22, No. 3, pp. 459-470, March 2003.
18. Ali Bilgin, George Zweig and Michael W. Marcellin, "Three-Dimensional Image Compression with Integer Wavelet Transforms," *Applied Optics*, Vol. 39, No. 11, pp. 1799-1814, April 2000.
19. YeongSeop Kim and William A. Pearlman, "Lossless Volumetric Image Compression," *Proc. Of SPIE, Applications of Digital Image Processing*, Vol. 3808, pp. 305-312, 2000.
20. Sungdae Cho, Dongyoun Kim and William A. Pearlman, "Lossless Compression of Volumetric Medical Images with Improved Three-Dimensional SPIHT Algorithm," *Journal of Digital Imaging*, Vol. 17, No. 1, pp. 57-63, March 2004.
21. J. Xu, Z. Xiong et al., "Memory-Constrained 3-D Wavelet Transform for Video Coding Without Boundary Effects," *IEEE Transactions on CSVT*, Vol.12, No.9, September 2002.
22. A. Golwelkar, "Motion Compensated Temporal Filtering and Motion Vector Coding using Longer Filters," *Ph.D. Dissertation*, Electrical, Computer and Systems Engineering Dept., Rensselaer Polytechnic Institute, September 2004.