LOSSLESS COMPRESSION OF 4D MEDICAL IMAGES USING H.264/AVC

V. Sanchez, P. Nasiopoulos and R. Abugharbieh

Department of Electrical and Computer Engineering, University of British Columbia

ABSTRACT

Four dimensional (4D) medical data are sequences of volumetric images captured in time. These data sets are typically very large in size and demand a great amount of resources for storage and transmission. In this paper, we present a lossless compression technique for 4D medical images which is based on the H.264/AVC video coding standard. Our lossless compression technique efficiently exploits spatial and temporal redundancies between 2D image slices and 3D images in 4D medical images and eliminates any concerns regarding the effects of compression on image quality for diagnostic purposes. Performance evaluations have shown that the proposed compression technique outperforms current 4D compression methods by 70%.

1. INTRODUCTION

Current medical imaging techniques, such as functional magnetic resonance (fMRI), positron emission tomography (PET), dynamic 3D ultrasound and dynamic computerized tomography (CT), produce large amounts of four dimensional (4D) data sets. These 4D medical data describe temporal changes of structures or processes (such as functional activation areas) as a sequence of three-dimensional (3D) images or volumes. Similarly, 3D medical images are a sequence of two dimensional (2D) image slices that represent cross sections of a structure. As medical images become more important in medical research and practice, the amount of multidimensional data information will consequently increase. Current improvements on data storage and communication techniques are helping reduce the cost of storing and transmitting large amounts of medical data. However, the growth rate of multidimensional medical image data is likely to surpass the rate of decrease in storage and bandwidth costs. Moreover, the increasing use of telemedicine, real-time tele-consultation and the Picture Archiving and Communications System (PACS), make medical image compression an important area of research. Therefore, designing cost effective and accurate compression schemes for efficient storage and transmission of the ever increasing spatial and temporal resolutions of modern medical image acquisition systems remains a challenge.

Compression algorithms may be broadly classified into lossy and lossless algorithms. While lossy techniques are able to achieve high compression ratios, they do not allow exact reconstruction of the original data. Due to the serious consequences of a possible erroneous diagnostic and its legal implications, medical images should always be stored in lossless format.

Most of the work done in the area of compression of multidimensional medical images is focused on 3D data [1,2,3]. Current 3D compression techniques treat each image slice independently (without exploiting the correlation in the third dimension), or compress the whole volume using 3D wavelet transforms. JPEG2000 and JPEG-LS are commonly used in the first case, and 3D-JPEG2000 in the latter [4,5]. Compression of 4D medical images, on the other hand, is still a new area of research [6]. Since 4D medical data can be represented as a set of 3D images, 3D compression algorithms may be used to encode each 3D image independently. Even though wavelet-based compression techniques appear to be a good choice to compress 4D medical image data, they do not exploit the redundancy among different volumes. One way to exploit the temporal redundancy between different volumes is by using motion compensation techniques which have been used successfully in video compression.

In this paper, we use motion compensation to losslessly compress 4D medical images. In particular, we use the H.264/AVC video standard, the latest and most advanced video coding method, to compress 4D functional magnetic resonance images (4D fMRI). Performance evaluations show that H.264/AVC provides a significance improvement in lossless compression compared to common compression techniques.

The rest of the paper is organized as follows. Section 2 briefly reviews 3D-JPEG2000. Our encoding method is described in Section 3. In section 4, we report the compression results and compare them to other compression techniques. Conclusions are presented in Section 5.

2. CODING 4D MEDICAL IMAGES

Four dimensional medical images are characterized by the high temporal redundancy among volumes. 3D wavelet-based compression algorithms try to reduce the redundancy among image slices by applying a wavelet transform along the three dimensions [4]. Among these 3D wavelet-based techniques, 3D-JPEG2000 has shown a good performance with 3D medical images [5].

3D-JPEG2000, as described in JPEG2000 Part 2, first applies a 1D wavelet transformation across image slices for the entire volume, and the resulting transformed slices are then encoded using JPEG2000. Fig. 1 shows the 3D wavelet decomposition specified by JPEG2000 Part 2.

Although, 3D-JPEG2000 may exploit the redundancies of 3D images, it fails to efficiently exploit the correlation in the fourth dimension of 4D data sets [6].

In the following section, we introduce our compression scheme for 4D medical images which is based on the H.264/AVC standard.

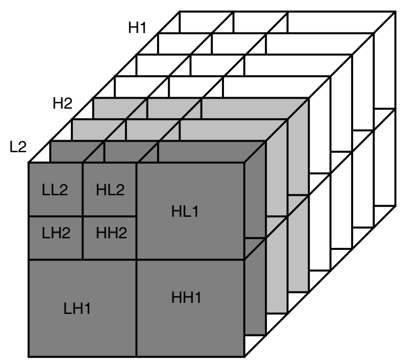

Fig. 1. 3D wavelet decomposition as specified in JPEG2000 Part 2. Illustration shows a 3D image with 8 image slices and two levels of decomposition.

3. OUR COMPRESSION SCHEME USING H.264/AVC

Consecutive image slices of 3D medical images have a significant amount of temporal redundancies and for this reason a 3D image may be treated as video sequence. 4D medical images may also be treated as video sequences, with each image slice of each volume representing a frame and the fourth dimension representing time. In this fashion, motion-compensation based compression may be applied to 4D medical images.

Motion compensation is an efficient way to reduce the temporal redundancy among different frames in a video sequence. The objective of the motion compensation process is to determine the amount of motion on a block by block basis which minimizes the difference between consecutive frames. H.264/AVC uses the most advanced and accurate motion estimate process and has proven to outperform any other video compression technique [7,8,9].

First, a picture is split into blocks. The first picture of a sequence is usually encoded as an "Intra" frame using only information contained in the picture itself. In H.264, even Intra-frames are obtained by using spatial prediction within a frame, a major departure from previous coding standards like MPEG-2 and MPEG-4 and even JPEG. The remaining pictures are typically encoded as "Inter" frames. H.264 uses motion compensation to generate Inter-frames that consist of a residual (which is the difference between the original and the predicted block) and motion vectors.

Variable motion estimation block sizes of 16x16, 16x8, 8x16 and 8x8 pixels are supported by H.264. In the case of 8x8, further partitions, which include 8x4, 4x8, or 4x4, might be used. Blocks with more motion details can be encoded using smaller block sizes. This improves the prediction by considering fine details, resulting in better compression rates than using fixed size blocks for motion estimation. H.264 also supports multi-picture motion-compensated prediction. Fig. 2 illustrates this concept.

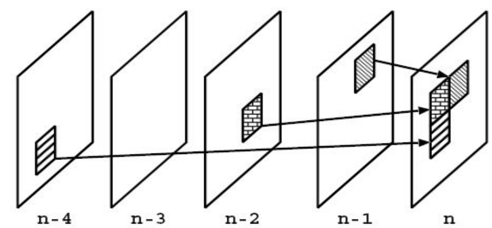

Fig. 2. Multi-frame motion compensation for H.264/AVC.

It has been shown that using multiple reference frames for prediction can yield 5-20% in bit rate savings as compared to using only one reference frame [7]. Multiple reference frames are especially beneficial for 4D medical images where image slices of subsequent volumes may have more temporal redundancies than two consecutive ones.

The residual information resulting from motion compensation or spatial prediction is transformed to frequency domain and the resulting coefficients are quantized. Variable length coding techniques are used to perform entropy coded. H.264 includes two unique methods for entropy coding, Context Adaptive Variable Length Coding (CAVLC) and Context Adaptive Binary Arithmetic Coding (CABAC) [3]. CAVLC has relatively less computational complexity and also includes Reversible Exp-Golomb codes to code some syntax elements. CABAC, however, offers better compression performance.

In the following sections we describe two encoding methods for lossless compression of 4D medical images. Both methods take advantage of the motion compensation model implemented in H.264/AVC. The first method, referred to as H.264-VOL hereafter, exploits the redundancy among image slices within each volume. The second method, referred to as H.264-TIME hereafter, exploits the

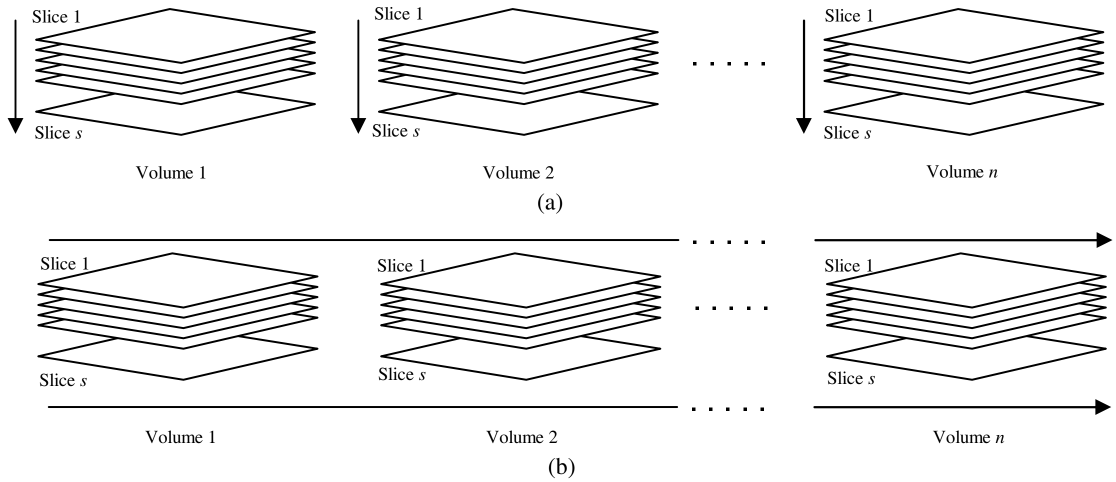

Fig. 3. Encoding order used by method H.264-VOL (a) and method H.264-TIME (b) for a 4D medical image with n volumes and s slices per volume.

similarities between image slices in the fourth dimension (time).

3.1. Method H.264-VOL

Method H.264-VOL exploits the redundancy among image slices within volumes of 4D medical images by applying H.264 along the third dimension. Since volumes in a 4D medical image describe cross sections of the same structure, the correlation between image slices may be efficiently exploited by using motion compensation. In order to achieve lossless compression, H.264-VOL bypasses the quantization process in H.264. After predicting all sample values of a block and calculating the residual of the prediction, the data is transformed and entropy coded using CABAC. In H.264-VOL, 4D medical images are treated as a set of volumes (3D images). Each volume is compressed separately. For instance, for Volume 1, image slices are fed to the encoder starting with the first image slice and ending with the last image slice of this volume. Therefore, each volume is encoded separately as a complete video stream and it may be decoded without having to decode the whole 4D data set. Fig. 3 (a) shows the encoding order used by H.264-VOL for a 4D medical image with n volumes, with s image slices per volume.

3.2. Method H.264-TIME

Similar to H.264-VOL, method H.264-TIME achieves lossless compression by bypassing the quantization process. However, unlike H.264-VOL, method H.264-TIME considers the fourth dimension of the 4D medical image. H.264-TIME applies H.264 across the fourth dimension (time) by grouping together all similar image slices across all volumes. Thus, the first group or “video stream” starts with the first image slice of the first volume, followed by the first image slice of the second volume and so on, and ends with the first image slice of the last volume. Therefore, each group of similar image slices is encoded separately and it may be decoded without having to decode the whole 4D data set. Since a 4D medical image describes the dynamic changes of a 3D image in time, the redundancies across the fourth dimension are higher than within 3D images. Therefore, H.264-TIME is expected to yield a better compression performance than the 3D case. Fig. 3 (b) shows the encoding order used by H.264-TIME for a 4D medical image with n volumes, with s image slices per volume.

4. EXPERIMENTAL RESULTS

We tested the compression performance of H.264-VOL and H.264-TIME using an 8-bit monochrome 4D fMRI data set consisting of 150 volumes. Each volume consists of 36 image slices with 128x128 pixels per image slice. We used the new Fidelity Range Extensions (FRExt) in H.264/AVC to losslessly encode the data as a monochrome sequence. In order to evaluate the results obtained using H.264/AVC, we also used JPEG-LS, JPEG2000 and 3D-JPEG2000 to losslessly compress the 4D fMRI data. JPEG-LS and JPEG2000 do not exploit the redundancy along the third and fourth dimensions. Therefore, each image slice was encoded independently when using JPEG-LS and JPEG2000. We used two levels of decomposition in JPEG2000.

We used two different approaches to compress the 4D fMRI data. The first approach compresses each volume independently. Using 3D-JPEG2000, we apply a 1D wavelet transform (with two levels of decomposition) along the third dimension of each volume of the 4D fMRI data. Next, all transformed slices are coded using JPEG2000 with two levels of decomposition. This approach is comparable to H.264-VOL, where each volume is compressed independently. Table 1 shows the compression ratios obtained by H.264-VOL, 3D-JPEG2000, JPEG2000 and JPEG-LS. We observe that 3D-JPEG2000 is better than JPEG2000 and JPEG-LS. This is expected since neither of the latter compression techniques takes into account the similarities among volumes. We also observe that H.264-VOL outperforms 3D-JPEG2000 by almost 25% when volumes are encoded independently. This confirms the advantages of H.264 and its motion compensation model even when the similarities among image slices within volumes are not high.

The second approach groups together all similar image slices along the forth dimension (i.e., time). Therefore, in this case, the first group of image slices consists of all the first image slices of each volume. For the case of JPEG2000 and JPEG-LS, there is no difference in compression. For 3D-JPEG2000, a 1D wavelet transform (two levels of decomposition) is applied to each group. Subsequently, all transformed slices are coded using JPEG2000 with two levels of decomposition. This approach is comparable to H.264-TIME, where similar image slices across the fourth dimension are grouped together and compressed independently. Table 2 shows the compression ratios for this second approach.

We observe that H.264-TIME clearly provides a better compression performance compared to 3D-JPEG2000. The improvement achieved by our approach is close to 70%. As expected, since the redundancy among the fourth dimension is higher, H.264 is capable to efficiently exploit this feature of 4D medical images, achieving higher compression ratios than the 3D approach.

Table 1. Compression ratios of 4D fMRI data using different lossless compression methods applied per volume.

Compression Technique	Compression ratio
JPEG2000	2.55:1
JPEG-LS	3.06:1
3D-JPEG2000	3.15:1
H.264-VOL	3.89:1

Table 2. Compression ratios of 4D fMRI data using different lossless compression methods applied across time.

Compression Technique	Compression ratio
JPEG2000	2.55:1
JPEG-LS	3.06:1
3D-JPEG2000	7.37:1
H.264-TIME	12.38:1

5. CONCLUSIONS

We proposed a lossless compression technique for 4D medical images. Our technique makes use of the H.264/AVC standard and efficiently exploits spatial and temporal redundancies in 4D medical images. Performance evaluations have shown that H.264/AVC significantly outperforms 3D-JPEG2000, reaching an impressive 70% improvement.

6. REFERENCES

[1] P. Schelkens, A. Munteanu, J. Barbarien, M. Galca, and X. Giro-Nieto, "Wavelet coding of volumetric medical datasets," *IEEE Transactions on Medical Imaging*, vol. 22, no. 3, pp. 441-457, March 2003.

[2] G. Menegaz and J. Thiran, "Lossy to lossless object-based coding of 3-D MRI data", *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 1053-1061, September 2002.

[3] H. J. Kim, J. H. Kim and C. W. Lee, "Lossless compression of 3D medical images using reversible integer wavelet transforms", *Proceedings of SPIE Int. Soc. Opt. Eng.*, vol. 4067, pp. 497-505, 2000

[4] J. Wang and H. K. Huang, "Medical image compression by using three-dimensional wavelet transformation", *IEEE Transactions of Medical Imaging*, vol. 15, no.4, pp. 547-554, August 1996.

[5] E. Siegel, K. Sidddiqui and J. Johnson, "Compression of multi-slice CT: 2D vs. 3D JPEG2000 and effects of slice thickness", *Proceedings of SPIE Int. Soc. Opt. Eng.,* vol. 5748, pp. 162-170, 2005.

[6] L. Zeng, C. Jansen, M. Unser and P. Hunziker, "Extension of wavelet compression algorithms to 3D and 4D image data: exploitation of data coherence in higher dimensions allows very high compression ratios", *Proceedings of SPIE Int. Soc. Opt. Eng.,* vol. 4478, pp. 427-433, 2001.

[7] A. Joch, F Kossentini and P. Nasiopoulos, "A performance analysis of the ITU-T draft H.26L Video Coding Standard", *Proceedings of the 12th International Packetvideo Workshop*, Pittsburgh, USA, Apr. 2002.

[8] ISO/IEC JTC 1/SC 29/WG 11 (MPEG), "Report of the formal verification tests on AVC/H.264", MPEG document N6231, December 2003.

[9] G. Sullivan, P. Topiwala and A. Luthra, "The H.264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions", *Proc. SPIE Int. Soc. Opt. Eng.*, vol 5558, pp. 454-474, August 2004.