

VOLUMETRIC COLOR IMAGE COMPRESSION USING SET PARTITIONING METHODS

Yi Mu

**Desert Research Institute
2215 Raggio Parkway
Reno, NV 89512
Tel: 775-673-7604
Fax: 775-673-7379**

Beddhu Murali, Junfeng Gu

**The University of Southern Mississippi
School of Computing
118 College Drive #5106
Tel: 601-266-4949**

Abstract - In this work, we present the applications of three-dimensional set partitioning methods to the sequence of still color images. The set partitioning methods we use in this paper are SPIHT, a state-of-the-art encoder and SPECK, a more recently developed, low complexity encoder. The three-dimensional versions of these methods are based on the observation that the sequences of images are contiguous in the temporal axis and there is no motion between slices. Therefore, the 3D discrete wavelet transform can fully exploit the inter-slices correlations. The set partitioning techniques involve a progressive "bitplane" coding of the wavelet coefficients, where the SPECK uses a cube-splitting quantization structure and the SPIHT uses a zerotree-like quantization structure. We extend the 3D-SPECK and 3D-SPIHT to code the color image sequences and call these schemes 3D-CSPECK and 3D-CSPIHT. Rate-distortion (Peak Signal-to-Noise Ratio (PSNR) vs. bit rate) performances were presented by comparing 3D-CSPECK and 3D-CSPIHT on one sequence of Visible Human datasets. Results show that 3D-CSPECK is comparable to 3D-CSPIHT, which matches the published results of gray scale image sequence compression.

Keyword: image compression volumetric data color image wavelet

1. INTRODUCTION

Sequences of images tend to demand enormous memory and storage capacities. For example, consider the Visible Human datasets from the National Library of Medicine. The cryosectional images of the Visible Female consist of images taken at one-third mm intervals resulting in a dataset of about 40 Gbytes. Access and transport of these data sets will stress existing processing, storage and transmission capabilities. Therefore, efficient compression should be applied to these data sets before storage and transmission.

Many promising image compression algorithm based on wavelet transform(WT)[1] were proposed recently. There are simple, efficient and have been widely used in many applications. Examples include the EZW (Embedded Zerotree Wavelet) algorithm [2], SPIHT (Set Partition In Hierarchical Trees) algorithm, the improved EZW algorithm of Said and Pearlman [3], and the SPECK (Set Partition Embedded Block) algorithm by [4], which offers comparable results to SPIHT with lower complexity.

SPIHT, as the state-of-the-art encoder, has many attractive properties. It is an efficient embedded technique. SPIHT has been extended to 3D by Kim and Pearlman [5] and it has been proved as a powerful tool to compress image sequences. 3D-SPIHT is the modern-day benchmark for three dimensional image compression.

In [6], Tang extends 2D SPECK to 3D sources and apply it to hyperspectral images. The proposed 3D-SPECK exploits detailed underlying physical modeling properties of hyperspectral images and produces comparable result to 3D-SPIHT in hyperspectral images compression.

Color pictures display more information than grayscale images since color pictures display the same number of grayscale tones as in grayscale images plus a number of colors on every image, thus, improving contrast resolution. From a medical point of view, color images disclose important information, such as tissue characterization, which is considered to be critical for diagnostic purposes.

In [7], Mu extends the 3D-EZW to color space and shows that 3D-CEZW is a powerful tool for color image sequences compression. In literature SPECK and SPIHT have been reported be superior than EZW [4], [5]. In this paper, we present the extension of the 3D-SPIHT and 3D-SPECK to color image sources, these state-of-the-art methods exploit the color space relationships as well as maintain the fully embeddedness required by color image sequences compression.

This paper is organized as following: we will describe the proposed algorithm in Section 2. Section 3 presents experimental results, and Section 4 concludes the paper.

2. COLOR THREE-DIMENSIONAL SET PARTITIONING TECHNIQUES (3D-CSPECK AND 3D-CSPIHT)

Many researchers already extended 2-D Set Partitioning techniques to three-dimensional sources [5], [6] [8], [9]. To do this, people usually take a 3-D wavelet transform on the image sequence, and apply the extend sorting and partitioning algorithm to encode the source. For example, 3D-SPIHT sorts coefficients along the paths of 3D trees (one pixel corresponds to eight direct descendant pixels) instead of 2-D trees (one pixel to four direct descendants). The correlation and dependence in all three dimensions can be exploited automatically. SPECK sorting is through block splitting, which can zoom in quickly to areas of high energy and code them first and therefore can exploit the presence of significant high frequency intra-band components.

The family of set partitioned methods is originally designed for gray scale images compression, to apply them to color images, we need to understand color space first. Color image usually is in RGB format, the RGB color spaces are highly correlated, and so transformation to a less correlated space is a requirement for efficient lossy compression. Using standard transformations, the original RGB images were transformed to code the YCC color space such that the luminance channel Y is stored as one byte for each pixel. On the other hand, the two chrominance channels are stored as one byte for each block of, say, $n \times n \times n$ pixels. Thus, for example, if $n = 2$, instead of using an average of 3 bytes (24 bits) per pixel, one would only be using 10 bits per pixel.

One possible application of set partitioned methods to a color image would be to code each color channel separately as is usually done in a conventional color image coder. Then, the generated bit-stream of each color space would be serially concatenated. The drawback of this approach is that one can not obtain full embeddedness and precise rate control, since the decoder needs to wait until the full bitstream arrives to reconstruct and display. Based on the work of [10], we proposed a method which treat all color planes as one unit at the coding stage, and generate one mixed bit-stream so that we can stop at any point of the bit-stream and reconstruct the color image of the best quality at the bit-rate. In addition, it will automatically allocate bits optimally among the color planes. We call the new color set portioned method as 3D-CSPECK or 3D-CSPIHT, according to which sorting algorithm we have used. The implementations of 3D-CSPIHT and 3D-SPECK are based on the QccPack [11].

3. RESULT

Table 1 and figure 1 show the comparative coding result of 3D-CSPIHT and 3D-CSPECK. The dataset is a $256 \times 256 \times 64$ subvolume form the Visible Female head section. The background of the images were removed to improve coding efficiency [7].

Table 1. Comparison between 3D-CSPECK and 3D-CSPIHT, CR is short for compression ratio

3D-CSPECK		3D-CSPIHT	
CR	PSNR (dB)	CR	PSNR (dB)
209.77	Y: 30.99	209.77	Y: 31.05
	Cb: 43.48		Cb: 43.50
	Cr: 39.10		Cr: 40.08
76.80	Y: 35.92	76.80	Y: 36.08
	Cb: 43.68		Cb: 43.68
	Cr: 42.87		Cr: 42.82

4. CONCLUSION

In this work, we have successfully adapted two state-of-the-art image coders, SPIHT and SPECK for color image sequences compression. Compared to gray scale images, a color image can divulge new information and is thus becoming more and more important for disease diagnosis. Unfortunately, color image compressions are under investigated. To the best of our knowledge, our work is the first ever published to use the most advanced techniques for volumetric color image compression. We hope that our framework will stimulate interests in standardizing 3D compression methods and in developing evaluating criteria for color volumetric data.



Fig. 1. Comparative coding results of Visible Human Female Anatomy slice avf1210a at compression ratio 76.80: Original(right), 3D-CSPIHT(center) and 3D-CSPECK(left),

5. REFERENCES

- [1] S.Mallat, "Multifrequency channel decompositions of images and wavelet models," in *IEEE Trans. Acoust., Speech, Signal Processing*, 1989, vol. 37, pp. 2091–2110.
- [2] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," in *IEEE Trans. Signal Processing*, 1993, vol. 41, pp. 3445–3462.
- [3] A. Said and W. A. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees," in *IEEE Trans. on Circuits and Systems for Video Technology*, 1996, vol. 6, pp. 243–250.
- [4] A. Islam, "An embedded and efficient low-complexity hierarchical image coder," in *Proc. SPIE Visual Comm. and Image Processing*, 1999, vol. 3653, pp. 294–350.
- [5] B. Kim and W. A. Pearlman, "An embedded wavelet video coder using three-dimensional set partitioning in hierarchical tree," in *IEEE Data Compression Conference*, 1997, pp. 251–260.
- [6] Xiaoli Tang, William A. Pearlman, and James W. Modestino, "Hyperspectral image compression using three-dimensional wavelet coding," in *Image and Video Communications and Processing*, 1999, vol. 3653, pp. 294–350.
- [7] Y.Mu, B.Murali, and A.L.Ali, "Embedded image coding scheme using zerotrees of wavelet for visible human dataset," in *Proceedings of Asilomar Conference on Signals, Systems, and Computers*. US Naval Postgraduate and IEEE, 2005, vol. 39.
- [8] M. Antonini, M. Barlaud, P. Mathiew, and I. Daubechies, "Image coding using wavelet transform," in *IEEE Trans. Image Processing*, 1992, vol. 1, pp. 205–220.
- [9] P. L. Dragotti, G. Poggi, and A.R.P.Ragozini, "Compression of multispectral images by three-dimensional spiht algorithm," in *IEEE Trans. on Geoscience and remote sensing*, 2000, vol. 38, pp. 416–428.
- [10] W.A.Pearlman, A.Islam, N.Nagaraj, and A. Said, "Efficient, low-complexity image coding with a set partitioning embedded block coder," in *IEEE Trans. Circuits and Systems for Video Technology*, 2004, vol. 14, pp. 1219–1235.
- [11] James E. Fowler, "Qccpack version 0.51 11-aug-2005," <http://qccpack.sourceforge.net>.