

Framework of A Hybrid Image Compression Algorithm for View-dependent Visualization

Yanshan Bian

Graduation school of the Academy of Equipment, Beijing, China

Email: bys02201@163.com

Lingda Wu, Chao Yang, and Bo Li

The Key Lab, the Academy of Equipment, Beijing, China

Email: wulingda@163.com

Abstract—This paper presents a framework of a hybrid method designed to decompress image blocks of tile-pyramid used in virtual environment especially for flight simulation. It combines two compression algorithms, namely DCT-based compression and WT-based one. Two estimator functions are proposed to choose between these two algorithms. To balance decompression performance against image quality, a selective mechanism for decompression method is introduced that depends on LOD level, image quality, bit rates and tile size. Our method is CPU-based and achieves real-time rendering performance with high image quality superior to pure DCT-based method and WT-based method.

Index Terms—DCT, Wavelet Transform, hybrid compression

I. INTRODUCTION

With the quick development of satellite sensors, nowadays it's very convenient to obtain high-resolution images, which can be used in a lot of commercial and non-profit virtual environment platforms, for instance the Google Earth, NASA World Wind, Autodesk Map 3D, 3DGeo and etc. However, just like every coin having two sides, this leads to the amount of image data increasing rapidly and largely. In the range of several hundred square kilometers, if one meter resolution satellite images are used, the data volume would reach TBs and it can affect the rendering efficiency and the real-time generation of 3D scene. It is a vital problem to treat mass data efficiently when those mass data is used in flight simulation. So data compression is a smart way to store and transmit the mass data at the inexpensive cost. Much work has been done on image compression and many an algorithm has been proposed from lossless to lossy ones. Lossless algorithms, such as Huffman encoding, run length encoding and arithmetic coding[1], can get exactly the same reconstitution image as the original one but the

compression ratio is very low. Lossy algorithms such as vector quantization [2], fractal coding[3], DCT(Discrete Cosine Transformation)-based coding[4], WT(Wavelet Transform)-based coding[5] and neural network coding[6] can obtain high compression ratio but maybe poor results.

Images used for view-dependent visualization in virtual environment are divided into tiles with the same size and different spatial resolution. Usually tile-pyramid model and linear quadtree tile index are utilized for managing those out-of-core data. Those compressed tiles data in sight of users must be decompressed real-time after be transmitting into the main memory. Unfortunately, the aforementioned platforms take the same compression algorithm for a given large image divided into small tiles. The advantages of different compression algorithms are not taken into account.

Inspired by Lucas Ammann[7] and Christian Dick[8], they present two different hybrid methods designed to render heightfield data on GPU with pure rasterization and ray-casting. Our motivation is to improve decompression efficiency in real-time visualization for virtual environment application. So a hybrid and view-dependent image compression algorithm framework is presented which combines two famous compression algorithms: DCT-based coding and WT-based coding.

The reminder of this paper is organized as follows. We first review existing methods for image compression. In Section 3, we present an estimator function for different compression methods and then our hybrid algorithm. Finally some results are shown in Section 4, before concluding in Section 5.

II. RELATED WORK

In the following two subsections, we respectively present DCT-based coding techniques and WT-based methods.

A. DCT-based Compression

The discrete cosine transform is a traditional method used in still image compression. It expresses an image in terms of cosine functions. Such transform is commonly used in JPEG compression standard. The discrete cosine

Manuscript received October 9, 2012; revised March 12, 2013.

transform for an $N \times N$ pixels image block $f(x, y)$ is defined as follows:

$$F(u, v) = c(u)c(v) \cdot \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \frac{\pi(2x+1)u}{2N} \cos \frac{\pi(2y+1)v}{2N} \quad (1)$$

where $x = 0, 1, \dots, N-1$, and $y = 0, 1, \dots, N-1$. $c(u)$ and $c(v)$ are defined by

$$c(u) = \begin{cases} \frac{1}{\sqrt{N}}, & u = 0 \\ \frac{2}{\sqrt{N}}, & u = 1, 2, \dots, N \end{cases} \quad (2)$$

$$c(v) = \begin{cases} \frac{1}{\sqrt{N}}, & v = 0 \\ \frac{2}{\sqrt{N}}, & v = 1, 2, \dots, N \end{cases} \quad (3)$$

In particular, $F(0,0)$ is known as the direct current coefficient (DC) and the remaining coefficients are called the alternating current coefficients. Most of the energy of the signal is packed in the DC coefficient.

DCT does not reduce the image amount before quantization is executed, and then Huffman encoding or other ones is performed to produce compressed bitstreams. Decompression is just a reverse of compression. DCT computation is very quick both on compression and decompression. But when it comes to low bpp(bits per pixel), block effect is obvious, as can be seen in Fig. 1(a).

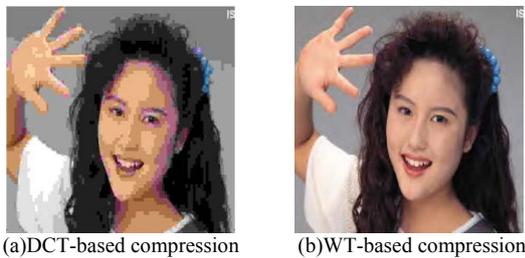


Figure 1. Reconstitution images where bpp is 0.125

B. WT-based Compression

Wavelet transforms involve representing a general function in terms of simple, fixed building blocks at different scales and positions. These building blocks are generated from a single fixed function called mother wavelet by translation and dilation operations. So wavelet transforms are capable of “zooming-in” on high frequency signal, and “zooming-out” on low frequency signal.

The purpose of wavelet transforms is to represent an image into the frequency-time domain. If $\psi(t) \in L^1 \cap L^2$ and $\hat{\psi}(0) = 0$, we can get function cluster $\{\psi_{a,b}\}$ by scaled and shifted of mother wavelet $\psi(t)$:

$$\psi_{a,b}(t) = |a|^{-1/2} \psi\left(\frac{t-b}{a}\right) \quad a, b \in R, a \neq 0 \quad (4)$$

For all $f \in L^2(R)$, the wavelet transform is of $f(x)$ can be defined as following:

$$(W_\psi f)(a, b) = |a|^{-1/2} \int_R f(x) \overline{\psi\left(\frac{t-b}{a}\right)} dt \quad (5)$$

This is the one-dimensional continuous formula and two-dimensional discrete wavelet transform(DWT) for images is computed as well as for $f(x)$. In theory, the original signal can be precisely reconstructed by the decomposed signal, but not all wavelet bases are suitable for image decomposition in image compression. The filter properties of corresponding wavelet bases have important relation with image compression. But this is out of the discussion range of this paper.

After wavelet transforms quantization algorithm must be done as DCT-based ones to realize image compression. Important wavelet coefficients are critical to reconstitute the original images and they should be quantized firstly. EZW[9], SPIHT[10], SPECK[11], and EBCOT[12] are some outstanding quantization methods. WT-based compression algorithms can realize discretional compression ratio and good quality. In compared to DCT-based ones, WT-based methods’ reconstituted images have splendid quality especially under low bit rate or bpp condition, as can be seen in Fig. 1(b).

III. HYBRID IMAGE COMPRESSION ALGORITHM

The hybrid image compression needs to decide whether to use DCT-based methods or WT-based methods, depending on which method can compress and decompress an image tile faster at the current view. For most of these two kinds of methods, they are symmetrical and the processing time of compression and decompression are almost the same. Therefore, for each compression method an estimator function that predicts the coding time for a particular tile in milliseconds(ms) is used. These estimator functions are parameterized by image tile and view dependent properties and they can be learned in training phase.

In the training phase, the hybrid algorithm picks a set of representative image tiles and views of the scene, and it renders these tiles using both DCT and WT. The measure rendering times are then used to determine the constants of the estimator functions.

The estimator functions are designed to be evaluated entirely on the CPU. The size of image tile is constant for virtual environment with tiled pyramid technology. To predict the time required to decode a tile using DCT-based coding algorithm we use the following estimator functions:

$$t_{DCT} \approx O(L, Q, S) \approx c_1 \cdot \frac{1}{L^{+1}} + c_2 \cdot Q \cdot S + c_3 \quad (6)$$

Here, L denotes the level of tiled pyramid(a sort of static LOD), Q denotes the quality(0~100) of image and S denotes the size of the image tile. c_1, c_2 and c_3 are CPU-specific constants (in ms).

The performance of WT-based compression algorithm is estimated by:

$$t_{WT} \approx O(L, B, S) \approx c_4 \cdot 2^{-L+1} + c_5 \cdot B \cdot S + c_6 \cdot S + c_7 \quad (7)$$

where B denotes the bit rate of compressed image, c_4, c_5, c_6 and c_7 are again CPU-specific constants (in ms). L is closely related to the distance from view point to the image tile when decompression is executed in virtual

environment exploring. And the other parameters are as the some as in (6). When we are close to a terrain block, high-resolution image is needed, so low LOD level tile compression bitstream is transmitted from storage to the main memory. And the *PSNR*(Peak Signal to Noise Ratio) of reconstituted image is proportional to *B*. If *PSNR* is larger than 30, people could not easily tell the difference between the original image and the reconstituted one. When user is far away from the terrain block, small *PSNR* with data loss is of tolerance. In our virtual environment application, *Q* and *B* are in inverse proportion to *L*. Based on a number of experiments we have made on real data sets, we set *S* to be constant for all tiles, reducing the above two estimator functions (6) and (7) to:

$$t_{DCT} \approx O(L) \approx c'_1 \cdot \frac{1}{L+1} + c'_3 \tag{8}$$

$$t_{WT} \approx O(L) \approx c'_4 \cdot 2^{-L+1} + c'_5 \cdot \frac{1}{L+1} + c'_6 \tag{9}$$

In theory, DCT-based compression algorithms are more suitable for low level of LOD tiles corresponding to high-resolution images. As we know, JPEG is quicker than JPEG2000 with the same conditions while they are respectively based on DCT and WT. But WT-based methods are better in quality with low bit rates or bpps.

WT-based methods can change the compression ratio arbitrarily and the quality of reconstituted images is dependent on people’s visual system mostly, and different LOD level should have different image resolution, so WT-based methods should change the LOD level and the bit rate many times until achieving the appropriate result consistent with the LOD level, as is depicted in Fig. 2. After the training phase, the estimator functions are related to LOD level, tile size and bit rate, when a image tile is ready for compression, estimator functions compare their compression time and choose the proper compression method. Then decompression flag and compression bit stream are stored for visual environment.

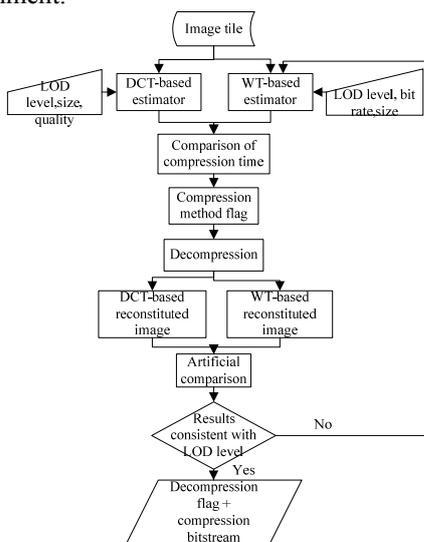


Figure 2. Diagram of our hybrid method in training phase

IV. RESULTS

In this paper, we choose JPEG as a representative for DCT-based compression algorithms and SPIHT(Set Partitioning In Hierarchical Trees) algorithm based on the second generation of discrete wavelet transform-Lifting Scheme on Daubechies 9/7 wavelet transform[10] as a representative for WT-based methods. The constants used in the estimator functions ((8) and (9)) for compression time are determined experimentally at the startup of the application by measuring the tiles’ DCT-based methods and WT-based methods for a number of randomly selected view positions and directions. The measurements for an Intel Core(TM)2 Duo CPU with 3GHz basic frequency and 2G memory are presented in Fig.3. The test images are 24-bit bitmaps and the size of each tile is 256×256 pixels.

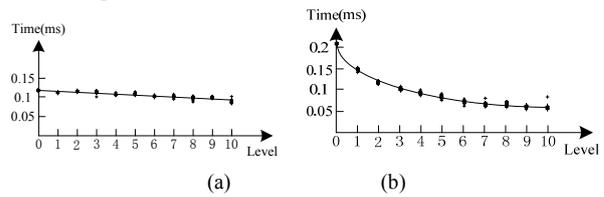


Figure 3. (a)DCT-based compression time vs. the LOD level.(b)WT-based compression time vs. the LOD level.

It shows in Fig. 3 the DCT-based compression time vs. the LOD level, and the WT-based compression time vs. the LOD level. To obtain the constants, a line and a curve are respectively fitted through the measurements, and the constants are derived from the line/curve parameters. At low level of LOD, DCT-based method consumes fewer time than WT-based one, this is decided by their algorithms. With the increasing of LOD level, the compression time of WT-based method decreases quickly while DCT-based one almost keeping the same as before. This is partly because the wavelet transform progress is nearly the same but the quantizations of coefficients are different with different bit rates or LOD level and it needs a lot computation for high LOD level.

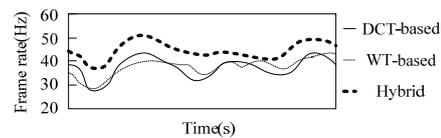


Figure 4. Speed rendering comparison between DCT-based method, WT-based method and our hybrid method.

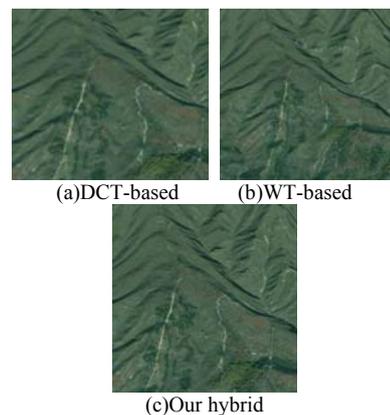


Figure 5. Rendering result comparison.

After training phase, all parameters are derived from experimental results and then they are used for virtual environment application platform. The open source OSGEarth is our experimental platform and we set the max LOD level to be twelve. As shown in Fig. 4, we are able to maintain real-time performances much superior to pure DCT-based method or WT-based one in most situations. In some case, for example, when the view point is very far from the terrain or is very close to the terrain, only one method is selected for decompression, so the performance of our hybrid method is almost equal to WT-based method or DCT-based algorithm.

In Fig. 5 it shows that our hybrid method has a good quality in the vicinity as DCT-based method while as WT-based method far away. DCT-based coding has advantages in low LOD level for its fast calculation and good quality. But the disadvantages show up at far view distance where low LOD level image tiles are needed, the right top "road" in Fig. 5(a) is blurry and we can hardly find it while the same "road" in Fig. 5(b) has a general outline. And our hybrid result in Fig. 5(c) shows a good quality from the near to the distant.

In this paper, we restrict our focus to the decompression efficiency while maintaining a acceptable result. This is very importance in flight simulation where rendering efficiency is highlighted.

V. CONCLUSIONS

In this paper, we have introduced a hybrid method designed to decompress image tiles with a good balance between speed and quality and flexibility. We combined two different compression methods: a DCT-based compression and a WT-based compression algorithm. To combine these methods, we proposed two estimator functions that compress image tiles according to LOD level which is dependent to view distance in order to maintain high decompressing performance without much visual quality loss.

Our compressing process only needs simple interactive pre-processing steps to obtain the constants of the two estimator functions. After that, image tiles are compressed into bitstream with DCT/WT flag.

We only study two kinds of compression algorithms and one method for each as the representative for experiments. There are also a lot of additional compression algorithms such as vector quantization, fractal coding and neural network coding. However our framework is open to all other compression algorithms and the estimator functions are needed to be redesigned.

Finally, in our future works, we also would like to study many other sorts of compression methods and make full use of their advantages, especially for different application requirements. And we believe an adaptive hybrid according to terrain complex and LOD level is well suited to compress and decompress images with a better balance between speed and quality.

REFERENCES

- [1] P.Y. Chen and J. M. Jou, "Adaptive arithmetic coding using fuzzy reasoning and grey prediction," *Fuzzy Sets and Systems*, vol. 114, pp. 239-254, September 2000.
- [2] Y. Lind, A. Buzo, and R. M. Gray, "An Algorithm for vector quantizer design," *IEEE Trans. on Communications*, vol. 28, pp. 80-95, January 1980.
- [3] A. E. Jacquin, "Fractal image coding based on a theory of iterated contractive image transformations," *SPIE Visual Communications and Image Processing*, September 1990, vol. 1360, pp. 227-239.
- [4] G. K. Wallace, "The JPEG still picture compression standard," *Communications of the ACM-Special Issue on Digital Multimedia Systems*, April 1991, vol. 34, pp. 30-44.
- [5] M. Boix and B. Cantó, "Wavelet Transform application to the compression of images," *Mathematical and Computer Modelling*, vol. 52, no. 7-8, pp. 1265-1270, February 2010.
- [6] V. Singh, N. Rajpal, and K. S. Murthy, "Neuro-wavelet based approach for image compression," *Computer Graphics, Imaging and Visualisation*, Bangkok, Thailand, August 14-17, 2007.
- [7] L. Ammann, O. Génevaux, and J. M. Dischler, "Hybrid rendering of dynamic heightfields using ray-casting and mesh rasterization," in *Proc. Graphics Interface*, Canada, 2010, pp.161-168.
- [8] C. Dick, J. Krieger, R. Westermann, "GPU-aware hybrid terrain rendering," in *Proc. IADIS Computer Graphics, Visualization, Computer Vision and Image Processing*, Germany, 2010, pp. 3-10.
- [9] A. Quafi, A. T. Ahmed, Z. Baair, N. Doghmane, and A. Zitouni, "Color image coding by modified embedded zerotree wavelet(EZW) algorithm," presented at 2nd Information and Communication Technologies, Damascus, Syria, October 16, 2006.
- [10] Q. Y. CAI, G. S. Wang, and Y. X. YU, "Research of still image compression based on daubechies 9/7 wavelet transform," presented at 2nd International Conference on Future Computer and Communication, Wuhan, China, May 21-24, 2010.
- [11] A. Islam and W. A. Pearlman, "An embedded and efficient low-complexity hierarchical image coder," in *Proc. SPIE 3653, Visual Communications and Image Processing*, USA, 1999, pp. 294-305
- [12] M. Rabbani and R. Joshi, "An Overview of the JPEG2000 still image compression standard," *Signal Processing: Image Communication*, vol. 17, no. 1, pp. 3-48, January 2002.



Yanshan Bian was born in 1986. He is a student at Key Lab. of the Academy of Equipment, Beijing, China. He is now working at the Ph.D. degree in virtual reality technology and virtual simulation in the Academy of Equipment, Beijing.



Lingda Wu was born in 1962. She is the Professor at Key Lab. of the Academy of Equipment. Her research interests are mainly in virtual reality technology and multi-media technology.



Chao Yang was born in 1981. He is the assistant researcher at Key Lab. of the Academy of Equipment. His research interests are mainly in virtual reality technology and virtual simulation.



Bo Li was born in 1982. He is the assistant researcher at Key Lab. of the Academy of Equipment. His research interests are mainly in system integration and network management.