Robust Invisible Watermarking of Volume Data Using the 3D DCT

[1]Yinghui Wu, [2]Xin Guan, [1]Mohan S. Kankanhalli, and [1]Zhiyong Huang
[1]Department of Computer Science, [2]Department of Computational Science
National University of Singapore, Singapore 117543
huangzy@comp.nus.edu.sg

1. Introduction

Digitizing of visual data has had a dual impact. Firstly, it has enabled fast and efficient storage, transfer and processing. Secondly, duplication and manipulation of data has also become very easy and undetectable. This can lead to violation of copyrights. One solution to this problem is to embed a secondary signal (watermark) into the visual data in an unobtrusive and robust manner.

Techniques to embed and retrieve such secondary information, or stamp (called watermark), that conveys some information about the intended recipient or the lawful owner of the original data, have been of considerable research interest [2, 6]. However, these works have been mostly confined to visual data such as digital images [7] and digital video [4]. Recently, there have been novel techniques proposed for watermarking 3D models such as arbitrary triangle meshes [1, 10, 8], NURBS curves and surfaces [9] and volume data [5]. In [5], a watermarking method is proposed based on the wavelet transform. With the same goal but different transform and methodology, the method proposed in this paper is more robust against common attacks.

In this paper, we propose a novel watermarking algorithm for 3D volume data based on the spread-spectrum communication technique which is invisible and robust. "Invisible" means that the 2D rendered image of this watermarked volume is perceptually indistinguishable from that of the original volume. "Robust" watermarking implies that the watermark is resistant to most intentional or unintentional attacks. We have implemented the algorithm and conducted experiments, showing that the watermark is invisible in the volume rendered 2D images. This is further confirmed by computing the Signal-to-Noise Ratio (SNR) and the Peak-Signal-to-Noise Ratio (PSNR) of the 3D watermarked volume data. We addressed different attacks listed as follows and put more emphasis on the first six attacks since they are more commonly performed by attackers. The experiments show that the watermarking scheme is robust.

- Geometrical distortions. Typical geometrical distortions include affine transformations (rotation by a constant angle, spatial scaling, etc.), cropping of a block of the volume, replacing a subset of the volume with another one, etc.
- Addition of a constant offset to voxel values.
- Addition of Gaussian or non-Gaussian noise.
- Linear filtering, such as low pass and high pass filtering, or non-linear filtering, such as median filtering.
- Local exchange of voxel slices.
- Digital-to-analog and analog-to-digital conversions.
- Collusion. A number of authorized recipients of the same volume data should not be able to collude and use the differently watermarked volumes to generate an unwatermarked copy of the volume.
- Forgery. A number of authorized recipients of the same volume data should not be able to collude and form a copy of watermarked volume with the valid embedded watermark of a person not in the group.

It must be noted that if a particularly strong attack manages to remove the watermark from the volume data, then the quality of its 2D volume rendered images should be sufficiently degraded so as to make this tampered data useless.

2. The Volume Watermarking Technique

Watermarking a volume data is essentially the process of altering the voxel values in a manner to ensure that a viewer of its volume-rendered image does not notice any perceptual change between the original volume rendering and the watermarked volume rendering. We utilize the spread-spectrum technique [3] in the frequency domain in order to achieve this effect. The purpose of utilizing the frequency domain is to make the watermark robust by hiding it in multiple frequencies. Assume that volume V that needs to be watermarked is of the size $n_x \times n_y \times n_z$. The basic scheme of our watermarking technique is outlined below:

0-7695-1007-8/01 $10.00© 2001 IEEE

1. A $4 \times 4 \times 4$ block-based 3D discrete cosine transform (DCT) transform [11] is applied to the volume V. The $4 \times 4 \times 4$ 3D DCT is computed using:

   $$F(u,v,w) =$$

   $$\frac{1}{2\sqrt{2}} C(u)C(v)C(w)[\sum_{x=0}^{3}\sum_{y=0}^{3}\sum_{z=0}^{3} f(x,y,z) *$$

   $$cos\frac{(2x+1)u\pi}{8} cos\frac{(2y+1)v\pi}{8} cos\frac{(2z+1)w\pi}{8}],$$

   where

   $$C(u), C(v), C(w) = \begin{cases} \frac{1}{2\sqrt{2}}, & \text{if } u,v,w = 0. \\ 1, & \text{otherwise}. \end{cases}$$

   Note that in our case $f(x,y,z)$ corresponds to the voxel values and $F(u,v,w)$ corresponds to the 3D DCT coefficients. The $4 \times 4 \times 4$ block-size has been chosen as a trade-off between the computational complexity of the transformation and the availability of sufficient frequencies to hide the watermark.

2. To embed the watermark information bits $a_j \in \{1, -1\}$ the bits are first spread by a large spread factor cr , called the chiprate [3]. For spreading the information, the bit pattern is repeated in a raster-scan order to tile the entire volume of size $n_x \times n_y \times n_z$. This improves its robustness to geometrical attacks such as cropping. The spreading provides spatial redundancy by embedding the information bits into cr number of voxels:

   $$b_i = a_j \quad \forall i = j \times K \qquad (1)$$

   and K varies from 1 to cr. The spread bits b_i are then modulated with a pseudo-random-noise (PN) sequence.

   $$p_i \text{ where } p_i \in \{-1, 1\}. \qquad (2)$$

   This forms the basic watermark sequence.

3. The modulated signal, i. e. the watermark sequence w_i where $w_i = b_i \cdot p_i$, forms a volume W of size $n_x \times n_y \times n_z$. This watermark volume W is also transformed into the frequency domain by using a $4 \times 4 \times 4$ block-based 3D DCT transform.

4. For every DCT block $b_i^V \in V$ and the corresponding DCT block $b_i^W \in W$, the corresponding coefficients are added to form a watermarked block $b_i^{V'} = b_i^V + b_i^W$ which constitute the watermarked volume V'_{DCT} in the frequency domain.

5. The 3D inverse DCT is performed on V'_{DCT} to obtain a $n_x \times n_y \times n_z$ size volume V'. The inverse 3D DCT is done using:

   $$f(x,y,z) =$$

   $$\frac{1}{2\sqrt{2}}[\sum_{u=0}^{3}\sum_{v=0}^{3}\sum_{w=0}^{3} C(u)C(v)C(w)F(u,v,w) *$$

   $$cos\frac{(2x+1)u\pi}{8} cos\frac{(2y+1)v\pi}{8} cos\frac{(2z+1)w\pi}{8}].$$

   This new volume V' is the watermarked volume data corresponding to the original volume data V.

For any given set of volume rendering parameters, the 2D image produced by volume rendering on V' will be perceptually indistinguishable from the 2D image produced using V. Since a pseudo-noise sequence is used for modulation, the watermark sequence is also noise-like which ensures that the watermark is difficult to detect, locate and manipulate without compromising on the quality of the corresponding volume-rendered images.

3. Watermark Detection

For detecting the existence of the watermark, the DCT-transformed original volume data V is subtracted from the DCT-transformed watermarked volume data $\hat{V}'$ (we use $\hat{V}'$ instead of V' because it may have been subjected to attacks) to obtain the residual volume data DCT coefficients, i.e. $V^r = \hat{V}' - V$. The 3D inverse DCT is performed on this residual data V^r to obtain the residual watermark sequence $\hat{w}_i$. This $\hat{w}_i$ is then analyzed by correlating it with the same pseudo-noise sequence that was used in the embedding phase where correlation can be understood as demodulation followed by summation over the correlation window. The correlation window for each bit is the chiprate. If the peak of correlation is positive, the corresponding watermark bit is $+1$ else it is -1. Considering one subset of the watermark values $\hat{w}_i$ over the correlation window where $i \in 1 \ldots cr$

$$s_j = \sum_{i=1}^{cr} p_i \cdot \hat{w}_i = \sum_{i=1}^{cr} p_i^2 \cdot b_i + \Delta, \qquad (3)$$

where Δ being the error term which can be due to intentional or unintentional attacks. But by choosing a large cr we have adequate redundancy and the summation can be approximated as :

$$s_j = \sum_{i=1}^{cr} p_i \cdot \hat{w}_i \approx cr \cdot a_j. \qquad (4)$$

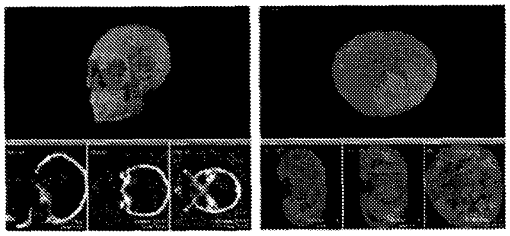
The required information bit $\hat{a_j}$ (i.e. the detected watermark bit) is

$$\hat{a_j} = sign(s_j). \qquad (5)$$

Thus, to retrieve the watermark, the original volume data and the same unshifted pseudo-noise sequence that was used at the embedder are required.

4. Test Results

In order to verify the robustness and invisible property of the algorithm proposed, we used a skull dataset (68 x 64 x 64) and a larger tomato dataset (64 x 208 x 216) to conduct tests (Fig. 1).



(a) The skull dataset (b) The tomato dataset

Figure 1. Two datasets used in algorithm evaluation (rendered by VolView).

After examining the watermarked volume after volume rendering, we found that the resultant watermarked volume of the algorithm above is indeed perceptually indistinguishable from the original (Fig. 2).

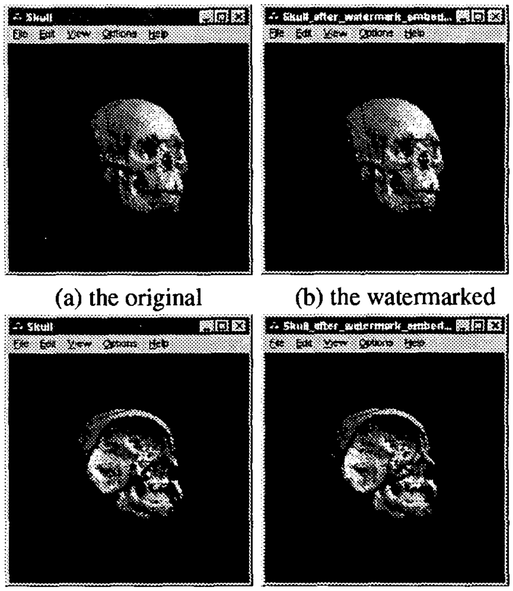
In order to examine the relationship between the robustness of the embedded watermark, we conducted a series of experiments using the same watermark signal sequence with different attacking strengths. The distortion caused by the attacks is measured in terms of SNR (Signal-to-Noise Ratio) and PSNR (Peak-Signal-to-Noise Ratio):

$$\begin{aligned} SNR &= 10log_{10}\frac{\sum_i v_i^2}{\sum_i (v_i' - v_i)^2}, \\ PSNR &= 10log_{10}\frac{\sum_i v_{max}^2}{\sum_i (v_i' - v_i)^2}, \end{aligned} \tag{6}$$

where v_i and v_i' are the ith-voxel values of the original and watermarked volume data respectively.

First, we conducted the cropping attacks. Experiment results of robustness on volume cropping is shown in Table 1 for the skull dataset. Watermark length $l = 1840$ bits and the Chiprate $cr = 53$.

We conducted tests of adding Gaussian noise. The attacks of resizing (with re-sampling), quantization and re-quantization, etc. can be modeled by signal noise addition. We are still trying to precisely model these attacks. We did the test of Table 2 using a 2D image of "NUS" logo as the watermark. The Fig. 3 shows the retrieved watermarks (2D images) under the various noise levels.


Skull
File Edit View Options Help
Skull_after_watermark_embed...
File Edit View Options Help
Skull
File Edit View Options Help
Skull_after_watermark_embed...
File Edit View Options Help

(a) the original (b) the watermarked

(c) the original (d) the watermarked

Figure 2. Comparison of volume-rendered images of the skull dataset before and after watermark embedding (rendered by PKVox).

For Table 3 of the tomato dataset, watermark length $l = 4000bits$ and the Chiprate $cr = 719$. Note that for digital images, noise with the PSNR higher than $35dB$ is hardly noticeable in general. A noise signal sequence with SNR less than $20dB$ is considered strong.

5. Conclusion

We have proposed a technique for three-dimensional volume data watermarking. A software package is implemented. We have also conducted tests and the results show that our method is robust.

References

[1] O. Benedens. Geometry-based Watermarking of 3D Models. *IEEE Computer Graphics & Applications.* Vol. 19, No. 1, 46-55, January/February 1999.

[2] L. Boney, A. H. Tewfik, K. Hamdy. Digital Watermarks for Audio Signals. *Proceedings of 1996 IEEE International Conference on Multimedia Computing and Systems (ICMCS '96).* Hiroshima Japan, 473-480, July 1996.

Cropping size	Remaining size (%)	SNR (dB)	PSNR (dB)	Error rate (%)
68 × 64 × 64	100.00	31.888	49.462	0.00
60 × 60 × 60	82.40	7.763	25.337	0.00
56 × 56 × 56	66.99	3.977	21.551	0.00
52 × 52 × 52	52.64	2.351	19.925	0.05
44 × 44 × 44	32.50	1.099	18.672	6.58
36 × 36 × 36	17.80	0.161	17.735	33.7

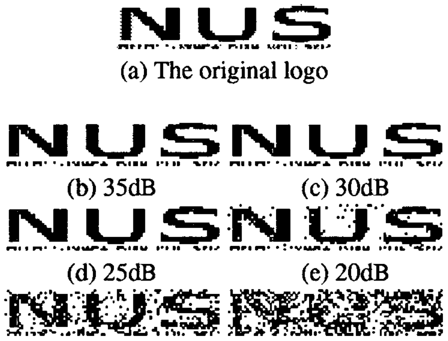
Table 1. Experiment results of robustness on volume cropping.

Gaussian noise SNR (dB)	SNR (dB)	PSNR (db)	Error rate (%)
35.0	30.1	47.9	0.00
30.0	28.4	46.2	0.00
25.0	25.7	43.5	0.11
20.0	21.7	39.5	4.10
15.0	17.2	35.0	16.0
10.0	12.4	30.2	30.0

Table 2. The skull dataset: experiment results of robustness on adding Gaussian noise.

Gaussian noise SNR (dB)	SNR (dB)	PSNR (db)	Error rate (%)
35.0	25.1	44.1	0.00
30.0	24.1	43.1	0.00
25.0	22.5	41.5	0.00
20.0	19.5	38.5	0.00
15.0	15.4	34.4	0.55
10.0	10.7	29.7	9.10
5.0	5.93	24.9	24.0

Table 3. The tomato dataset: experiment results of robustness against adding Gaussian noise.



(a) The original logo

(b) 35dB (c) 30dB

(d) 25dB (e) 20dB

(f) 15dB (g) 10dB

Figure 3. Retrieved watermarks shown as images under different SNR values (dB) of Gaussian noise.

[3] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon. Secure Spread-Spectrum Watermarking for Multimedia. *IEEE Transactions on Image Processing.* Vol. 6, No. 12, 1673-1687, December 1997.

[4] J. Dittman, M. Stabenan, R. Steinmetz. Robust MPEG Video Watermarking Technologies. *Proceedings of ACM MM '98.* Bristol, U.K., 71-80, September 1998.

[5] X. Guan, Y. Wu, M. S. Kankanhalli, Z. Huang. Invisible Watermarking of Volume Data Using Wavelet Transform. *Proceedings of MMM '2000.* Nagano, Japan, 152 - 166, Nov 2000.

[6] F. Hartung, M. Kutter. Multimedia Watermarking Techniques. *Proceedings of the IEEE.* Vol. 87, No. 7, 1079-1107, July 1999.

[7] M. S. Kankanhalli, Rajmohan, K. R. Ramakrishnan. Content-based Watermarking of Images. *Proceedings of ACM MM '98.* Bristol, U.K., 61-70, September, 1998.

[8] R. Ohbuchi, H. Masuda, and M. Aono. Watermarking three-dimensional polygonal models through geometric and topological modifications. *IEEE Journal on Selected Areas in Communications.* 16(14):551–560, May 1998.

[9] R. Ohbuchi, H. Masuda, and M. Aono. A shape-preserving data embedding algorithm for NURBS curves and surfaces. *Proceedings of CGI '99.* 180–187. IEEE Computer Society, 1999.

[10] E. Praun, H. Hoppe, A. Finkelstein. Robust Mesh Watermarking. *Computer Graphics ACM SIGGRAPH '99 Proceedings.* 69-76, September 1999.

[11] K. Rao, R. Yip. Discrete Cosine Transform: algorithms, advantages, applications. *Academic Press Inc.* 1990.