Reversible Watermarking by Modulation and Security Enhancement

Hao-Tian Wu and Yiu-Ming Cheung, *Senior Member, IEEE*

***Abstract*—This paper presents a watermarking algorithm that is suitable for the data represented in floating- or fixed-point numbers. By keeping the modulation information, every value in the original object can be restored with the smallest error. For the sake of exact recovery, the redundancy in the watermarked object should be exploited to save the recovery information. We analyze the security of the algorithm and enhance it by adding a pseudorandom (PR) sequence to the watermarked object. The algorithm can be applied in the measurement to embed some useful information in a distortion-free way. We implement it on the Virtual Reality Modeling Language (VRML) models, and the experimental results show its efficacy.**

***Index Terms*—Authentication, modulation, reversible watermarking, security, 3-D geometry.**

I. Introduction

In the past decades, digital watermarking techniques have received much attention in the community (e.g., [1]–[6]) for a variety of applications, such as copyright protection [4], authentication [5], quality evaluation [6], etc. Compared with cryptographic algorithms [7], fragile watermarking does not change the file format or affect the perceptual quality of a host (also called original object hereinafter), in which a tamperproof watermark is embedded. By comparing the extracted watermark with the embedded watermark, the watermarked object can be authenticated. In the research area of fragile watermarking, an attractive topic is reversible data hiding. Normally, a certain degree of distortion will definitely be introduced to the original object to generate the watermarked object. With the reversible watermarking technique, which is also referred to as lossless, distortion-free, or invertible hiding, we can revert to the exact copy of the original object by removing the distortion caused by the embedding. The property of reversibility is desirable particularly in high-precision applications, such as military, medical, as well as in the measurement.

In the literature, quite a few algorithms have been proposed to accomplish the property of reversibility. The typical methods are based on modulo 256 addition [8], lossless multiresolution transform [9], lossless compression [10], [11], invertible noise adding [12], difference expansion [13], [14], integer wavelet transform [15], circular interpretation of bijective transformation [16], histogram modification [17], and so forth. In general, these algorithms focus on digital images that are stored as integers from 0 to 255. How to reversibly embed information into the objects represented by floating- or fixed-point numbers, such as the 3-D models consisting of coordinates (e.g., [18]) and the high-dynamic-range images [19], has seldom been investigated. However, reversible hiding in any type of data is desirable to avoid information loss. Since most of the existing methods take advantage of the characteristics of digital images, directly applying them to the objects represented by floating- or fixed-point numbers may encounter difficulties or cause a large distortion. In our preliminary work [20], the idea of keeping the modulation information in the watermarked object is adopted in quantization-based embedding. In addition, we implement it on 3-D mesh models so that the original mesh model can approximately be recovered. The condition for the exact recovery is given in [21], and the recovery process can be performed without any specific information of the original object. Nevertheless, it is possible to estimate the quantizer employed in the modulation by the statistical analysis of the watermarked object, as shown in the following section. Therefore, the security of the algorithm needs to be enhanced to prevent the possible information leakage from the watermarked object.

We regard the security of reversible watermarking as the uncertainty and computational difficulty to correctly extract the embedded watermark and recover the original object without authorization. To enhance the security, a pseudorandom (PR) number generator is employed with a secret key as the seed. Then, a PR sequence is generated and added to the watermarked object to make it hard to estimate the employed quantizer. In the recovery process, the added sequence should be subtracted from the watermarked object to extract the embedded watermark and recover the original object. The applicability of the watermarking algorithm is discussed, and it is interpreted as a kind of LSB hiding for the data represented in floating- or fixed-point numbers. For the sake of exact recovery, the methods to exploit the redundancy in the watermarked object are proposed to save the recovery information. To apply our algorithm in the measurement, we take the 3-D geometry with coordinates for instance, and the experimental results on the virtual reality modeling language (VRML) models [22] are given.

The rest of this paper is organized as follows. In Section II, we first introduce a watermarking algorithm by quantization-based modulation. Then, its security is analyzed and enhanced

Manuscript received August 30, 2007; revised November 14, 2008. First published August 18, 2009; current version published December 9, 2009. This work was supported in part by the Faculty Research Grant of the Hong Kong Baptist University under Project FRG/06-07/II-07 and in part by the Research Grant Council of Hong Kong SAR under Grant HKBU 2156/04E and Grant HKBU 210306. The Associate Editor coordinating the review process for this paper was Dr. Antonios Tsourdos.

H.-T. Wu is with the School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, China (e-mail: whaot@mail.sysu.edu.cn).

Y.-M. Cheung is with the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong (e-mail: ymc@comp.hkbu.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2009.2022453

by adding a PR sequence to the watermarked object. In Section III, the applicability of the watermarking algorithm is discussed. Moreover, its application in the measurement is illustrated in Section IV. The experimental results on the VRML models are given in Section V. Finally, we draw a conclusion in Section VI.

II. Security-Enhanced Watermarking Algorithm by Modulation

In [21], a quantization-based watermarking algorithm is introduced. First, a string of bit values are embedded into the original object based on the odd–even embedding, which is the simplest example of the quantization index modulation scheme [2]. Then, the modulation information is calculated and kept in the watermarked object. The general procedure can be summarized as follows.

A. Quantization-Based Modulation

To embed a string of bit values $\mathbf{W} = \{w_1, w_2, \ldots, w_N\}$ into an original object $\mathbf{Y} = \{Y_1, Y_2, \ldots, Y_N\}$ consisting of N values, there are two parts of information that need to be hidden in the watermarked object $\mathbf{Z} = \{Z_1, Z_2, \ldots, Z_N\}$. One part is the embedded data, i.e., $\mathbf{W}$, and the other part is the modulation information, which is the difference between the original object $\mathbf{Y} = \{Y_1, Y_2, \ldots, Y_N\}$ and the object $\mathbf{Y}' = \{Y_1', Y_2', \ldots, Y_N'\}$ generated from $\mathbf{Y}$, $\mathbf{W}$, and the quantization step size Δ. Further, a new signal $\mathbf{E} = \{e_1, e_2, \ldots, e_N\}$ associated with a parameter a is generated to represent the modulation information by $e_i = (Y_i - Y_i')/a$ for $i = 1, \ldots, N$. The detailed procedure is as follows.

Step 1) For $i = 1, \ldots, N$, initialize the integer quotient Q_i by $Q_i = \lfloor Y_i/\Delta \rfloor$ with the quantization step size Δ, where $\lfloor \cdot \rfloor$ represents the floor function. We define the function $B(\cdot)$ and the remainder R_i by

$$\begin{cases} B(Q_i) = Q_i - \left\lfloor \frac{Q_i}{2} \right\rfloor \times 2 \\ R_i = Y_i - Q_i \times \Delta \end{cases}. \tag{1}$$

It can be seen that the output $B(Q_i)$ is a bit value of 0 or 1, and the value of R_i is nonnegative. For example, $B(-7) = -7 - \lfloor -7/2 \rfloor \times 2 = -7 - (-4) \times 2 = 1$.

Step 2) A bit value w_i is embedded into Y_i' based on the odd–even embedding by

$$Y_i' = \begin{cases} Q_i \times \Delta + \frac{\Delta}{2}, & \text{if } B(Q_i) = w_i \\ Q_i \times \Delta - \frac{\Delta}{2}, & \text{if } B(Q_i) \neq w_i \ \&\ R_i \leq \frac{\Delta}{2} \\ Q_i \times \Delta + \frac{3\Delta}{2}, & \text{if } B(Q_i) \neq w_i \ \&\ R_i > \frac{\Delta}{2} \end{cases} \tag{2}$$

so that $Y_i' = \lfloor Y_i'/\Delta \rfloor \times \Delta + (\Delta/2)$, and $B(\lfloor Y_i'/\Delta \rfloor) = w_i$. Subsequently, $e_i = (Y_i - Y_i')/a$ is obtained.

Step 3) For $i = 1, \ldots, N$, e_i is further added to Y_i' by

$$Z_i = Y_i' + e_i = \left\lfloor \frac{Y_i'}{\Delta} \right\rfloor \times \Delta + \frac{\Delta}{2} + e_i. \tag{3}$$

If the difference between Y_i and Y_i' is denoted by γ_i, then it can be seen from (2) that $\gamma_i \in (-\Delta, \Delta]$. Since $e_i = \gamma_i/a$ for $i = 1, \ldots, N$, the signal $\mathbf{E} = \{e_1, e_2, \ldots, e_N\}$ will be distributed within $(-(\Delta/a), (\Delta/a)]$ if the parameter a is a positive value. Further, the parameter a should be assigned with a value greater than 2 so that $e_i \in (-(\Delta/2), (\Delta/2))$ for $i = 1, \ldots, N$. Hence, the adding of e_i in (3) will not change the embedded value w_i because $B(\lfloor Z_i/\Delta \rfloor) = B(\lfloor Y_i'/\Delta \rfloor) = w_i$. Therefore, it is necessary to divide the modulation information, i.e., the difference between $\mathbf{Y}$ and $\mathbf{Y}'$, by a value greater than 2 so as to keep it in $\mathbf{Z}$ without changing the embedded data $\mathbf{W}$.

B. Watermark Extraction and Recovery Process

With the quantization step size Δ, the embedded bit values $\mathbf{W} = \{w_1, w_2, \ldots, w_N\}$ can be extracted from the watermarked object $\mathbf{Z} = \{Z_1, Z_2, \ldots, Z_N\}$ by

$$w_i = B\left(\left\lfloor \frac{Z_i}{\Delta} \right\rfloor \right). \tag{4}$$

From a practical point of view, the precision of the watermarked object $\mathbf{Z}$ needs to be taken into account. Supposing that $\mathbf{Z}$ is stored at the precision level of 10^{-m}, the roundoff error is within $(-5 \times 10^{-(m+1)}, 5 \times 10^{-(m+1)}]$. To ensure that the value of $\lfloor Z_i/\Delta \rfloor$ is not affected so that w_i can correctly be extracted, the following condition should be satisfied:

$$0 \leq \frac{\Delta}{2} + e_i \pm 5 \times 10^{-(m+1)} < \Delta \tag{5}$$

which is equivalent to $(\Delta/2) - (\Delta/a) > 5 \times 10^{-(m+1)}$ as $e_i \in (-(\Delta/a), (\Delta/a)]$. Given $a > 2$, the embedded data can correctly be extracted if

$$\Delta > \frac{a}{a-2} 10^{-m}. \tag{6}$$

In practice, Δ should be as small as possible to minimize the distortion caused by data embedding. Given the original object $\mathbf{Y}$ at the precision level of 10^{-n} and $n = m$, the value of Δ should be greater than $(a/(a-2))10^{-n}$.

To recover the original object $\mathbf{Y} = \{Y_1, Y_2, \ldots, Y_N\}$, the modulation information $\mathbf{E} = \{e_1, e_2, \ldots, e_N\}$ should be retrieved from $\mathbf{Z} = \{Z_1, Z_2, \ldots, Z_N\}$. For $i = 1, \ldots, N$, we have

$$e_i = Z_i - \left(\left\lfloor \frac{Z_i}{\Delta} \right\rfloor \times \Delta + \frac{\Delta}{2} \right) = Z_i - Y_i' \tag{7}$$

where $Y_i' = \lfloor Z_i/\Delta \rfloor \times \Delta + (\Delta/2)$ can easily be obtained from Z_i with the quantization step size Δ. Subsequently, for $i = 1, \ldots, N$, a value Y_i in $\mathbf{Y}$ can be obtained by

$$Y_i = Y_i' + e_i \times a. \tag{8}$$

If the watermarked object $\mathbf{Z}$ is stored at the same precision as the original object $\mathbf{Y}$, then only a part of the modulation information can be saved because of the roundoff error. Given $a > 2$ and (6), it can be seen from (8) that the error introduced to the recovered object will be within $(-5a \times 10^{-(m+1)},$

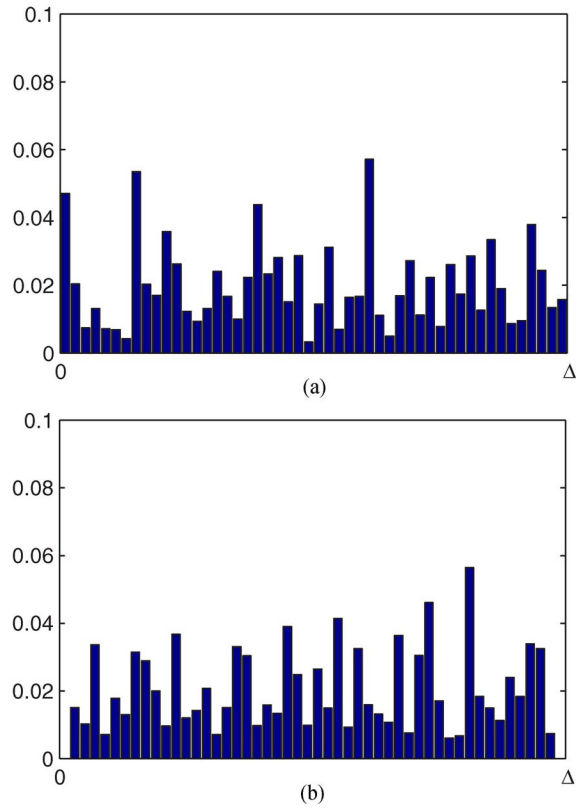

Fig. 1. PMFs of R_i and $(\Delta/2)+e_i$ for the coordinates in the original and watermarked VRML models of "indigo", respectively, with $a=2.1$ and $\Delta=0.0013$. (a) The PMF of R_i for the VRML model of "indigo". (b) The PMF of $(\Delta/2)+e_i$ for the watermarked model.

$5a\times 10^{-(m+1)}]$. The difference between the recovered and original objects may be out of the range $(-5\times 10^{-(m+1)}, 5\times 10^{-(m+1)})$ so that the recovered object is unequal to the original object. Nevertheless, the range of the introduced error can be controlled by the parameter a. For instance, if we set a within (2, 3), then the difference between the recovered object and the original object will be within $(-1.5\times 10^{-m}, 1.5\times 10^{-m})$ so that the possible error is either 10^{-m} or -10^{-m}.

C. Security Enhancement

In the proposed algorithm, the embedded watermark and the modulation information may be disclosed if the quantization step size Δ is known. In the following, we show that it is possible to estimate the quantizer employed in the modulation by the statistical analysis of the watermarked object. As shown in Section II-A, $e_i\in(-(\Delta/a),(\Delta/a)]$ so that $(\Delta/2)+e_i$, and the remainder of Z_i modulo Δ by constraining the quotient to be an integer is within $((\Delta/2)-(\Delta/a),(\Delta/2)+(\Delta/a)]$. By assigning a value greater than 2 to a, there are a couple of gaps in the probability mass function (PMF) of $(\Delta/2)+e_i$. More accurately, $(\Delta/2)+e_i$ will not be distributed in the intervals of $[0,(\Delta/2)-(\Delta/a))$ and $((\Delta/2)+(\Delta/a),\Delta)$. By setting $a=2.1$ and $\Delta=0.0013$, the PMFs of the remainders of the coordinates in the original and watermarked models of "indigo" modulo Δ (i.e., R_i and $(\Delta/2)+e_i$) are shown in Fig. 1, respectively. We calculate the PMF by dividing the interval of $[0,\Delta)$ into 50 subintervals with the same size, counting the numbers of R_i and $(\Delta/2)+e_i$ in every interval and normalizing the obtained numbers. For the values in an original object such as the VRML model of "indigo", there is no characteristic gap in the PMF of R_i, as shown in Fig. 1(a). However, as shown in Fig. 1(b), none of the remainders of the values in the watermarked model modulo Δ, i.e., $(\Delta/2)+e_i$, is distributed in the first and last subintervals because $(1/2)-(1/a)=(1/2)-(10/21)>(1/50)$ and $(1/2)+(1/a)=(1/2)+(10/21)<(49/50)$. From the PMF of $(\Delta/2)+e_i$, we can easily find the gaps and estimate the value of Δ. Noting that 2.1 is only slightly greater than 2, the gaps will become more obvious if the value of a is increased. Therefore, information leakage is possible if no protection measure has been made.

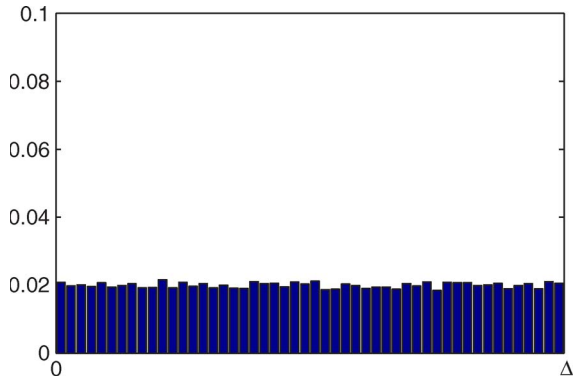

Fig. 2. With $a=2.1$ and $\Delta=0.0013$, the PMF of $((\Delta/2)+e_i+V_i)\%\Delta$ for the coordinates in the watermarked model of "indigo'" after a PR sequence $\mathbf{V}$ is added.

To enhance the security, a PR number generator is employed with a secret key K as the seed to generate a PR sequence $\mathbf{V}=\{V_1,V_2,\ldots,V_N\}$. For consistency, we use $\mathbf{Z}'=\{Z'_1,Z'_2,\ldots,Z'_N\}$ to denote the object obtained in Section II-A instead of $\mathbf{Z}=\{Z_1,Z_2,\ldots,Z_N\}$. Inspired by the technique of dither modulation [2], the PR sequence $\mathbf{V}$ is added to $\mathbf{Z}'$ to produce the final watermarked object $\mathbf{Z}=\{Z_1,Z_2,\ldots,Z_N\}$ by

$$Z_i=Z'_i+V_i=\left\lfloor\frac{Z'_i}{\Delta}\right\rfloor\times\Delta+\frac{\Delta}{2}+e_i+V_i \qquad (9)$$

for $i=1,\ldots,N$. By making V_i randomly distributed within the interval of $(-(\Delta/2),(\Delta/2))$, the gaps in the PMF of $(\Delta/2)+e_i$ can be filled. As shown in Fig. 2, the remainder of Z_i modulo Δ (i.e., $((\Delta/2)+e_i+V_i)\%\Delta$) for the coordinates in the watermarked model of "indigo" is close to uniform distribution in the interval of $[0,\Delta)$. Since it is hard to estimate the value of Δ from $\mathbf{Z}=\{Z_1,Z_2,\ldots,Z_N\}$, less information of the employed quantizer can be learned from the watermarked object. Furthermore, neither the embedded bit value w_i nor the modulation information e_i can be extracted from Z_i unless V_i has been subtracted. Therefore, the security of the watermarking algorithm has been enhanced by the adoption of a PR sequence.

As the PR sequence $\mathbf{V}$ is generated at the same precision as $\mathbf{Z}'$ in the embedding process, it should be at the same precision as the watermarked object $\mathbf{Z}$ in the recovery process, also with the secret key K as the seed of the PR number generator.

The object $\mathbf{Z}'$ obtained in Section II-A can be produced by subtracting $\mathbf{V}$ from $\mathbf{Z} = \{Z_1, Z_2, \ldots, Z_N\}$ by

$$Z'_i = Z_i - V_i = \left\lfloor \frac{Z'_i}{\Delta} \right\rfloor \times \Delta + \frac{\Delta}{2} + e_i \tag{10}$$

for $i = 1, \ldots, N$. After that, the embedded watermark $\mathbf{W}$ can be extracted from $\mathbf{Z}'$, and the original object $\mathbf{Y}$ can be recovered. If the embedding and recovery processes are performed by two different parties, then the key K, the parameter a, and the step size Δ need to be transferred (e.g., by using the public key infrastructure).

III. Applicability of the Watermarking Algorithm

If the watermarked object $\mathbf{Z}$ is at the same precision as the original object $\mathbf{Y}$, then the modulation information saved in $\mathbf{Z}$ is insufficient for the exact recovery of $\mathbf{Y}$. Nevertheless, the proposed algorithm is applicable in the following cases.

Case 1) The slight difference between the recovered object and the original object can be neglected. When $\mathbf{Z}$ is at the same precision as $\mathbf{Y}$, the error for each value is either -10^{-n} or 10^{-n} by setting the parameter a within (2, 3), where 10^{-n} is the precision level of $\mathbf{Y}$. Under these circumstances, the watermarking algorithm can be interpreted as a kind of the LSB hiding in a sense that only the minimum distortion may be introduced to every value. For the data represented in floating- or fixed-point numbers, the slight error may be tolerable. As for the data obtained in the measurement, the least significant digit is an estimated value.

Case 2) The algorithm is also applicable to the host where the errors can be saved. As for $2 < a < 3$, we can use -1, 0, and 1 to represent the errors -10^{-n}, 0, and 10^{-n}, respectively. With the error vector that can be obtained in advance, the original object $\mathbf{Y}$ can exactly be recovered. Given that the host is represented by floating-point numbers, lossless compression techniques can be employed to make room for the extra bits. For each floating-point number in the *single* precision (see the IEEE 754 standard [23]), we can take the two least significant bytes in the mantissa part. After we compress the chosen bytes with some lossless algorithm, the error vector can be appended. As a result, the file size of the watermarked object $\mathbf{Z}$ will not be increased, given that enough space can be saved by lossless compression for the error vector. A certain degree of distortion will be caused by replacing the two bytes in every floating-point number with the compressed bit stream and the error vector. However, the distortion will be removed by decompression in the recovery process.

Case 3) Another way is to increase the precision of the watermarked object $\mathbf{Z}$. After storing every value in $\mathbf{Z}$ with one more decimal digit, the precision level of $\mathbf{Z}$ becomes $10^{-(n+1)}$. Given $a \in (2, 3)$ and (6), the difference between the recovered and original objects is within the interval of $(-2 \times 10^{-(n+1)}, 2 \times 10^{-(n+1)})$. Therefore, the original object $\mathbf{Y}$ can exactly be recovered at the precision level of 10^{-n} after rounding. This method can only be used provided that there is redundancy in data representation; thus, the file size will not be changed by increasing the precision (e.g., both of the two values 1.652 and 1.6523 can be represented by floating-point numbers in single precision). Otherwise, the file size will also be increased when the precision is increased, such as the case that $\mathbf{Y}$ is represented by fixed-point numbers.

TABLE I
Procedure of the Proposed Watermarking Algorithm

The embedding process:
Assign a value within $(2, 3)$ to the parameter a;
Choose the quantization step-size Δ according to the precision of the original object $\mathbf{Y}$;
For each *value* Y_i *in* $\mathbf{Y} = \{Y_1, Y_2, \ldots, Y_N\}$ **do**
  Calculate the integer quotient Q_i so as to obtain the values of $B(Q_i)$ and R_i by Eq.(1);
  Embed a bit value w_i in Y'_i by Eq.(2);
  Obtain the modulation information $e_i = \frac{Y_i - Y'_i}{a}$;
  Add e_i to Y'_i by Eq.(3) to generate a value Z'_i in $\mathbf{Z}' = \{Z'_1, Z'_2, \ldots, Z'_N\}$;
Increase the precision of $\mathbf{Z}'$ (Case 3);
A PR sequence $\mathbf{V} = \{V_1, V_2, \ldots, V_N\}$ is generated at the same precision as $\mathbf{Z}'$ with a secret key K;
Add $\mathbf{V}$ to $\mathbf{Z}'$ to get the watermarked object $\mathbf{Z}$ by Eq.(9);
Compress $\mathbf{Z} = \{Z_1, Z_2, \ldots, Z_N\}$, then append the error vector to the compressed bitstream (Case 2);

The recovery process:
Retrieve the error vector (i.e. the appended bits) and decompress the watermarked object $\mathbf{Z}$ (Case 2);
Generate the PR sequence $\mathbf{V}$ with the key K;
Obtain $\mathbf{Z}'$ by subtracting $\mathbf{V}$ from $\mathbf{Z}$ with Eq.(10);
For each *value* Z'_i *in* $\mathbf{Z}'$ **do**
  Extract the embedded bit value w_i by Eq.(4);
  Retrieve the value of e_i by Eq.(7);
  Recover a value Y_i in $\mathbf{Y}$ by Eq.(8);
Exactly recover $\mathbf{Y}$ with the error information (Case 2);
Round $\mathbf{Y}$ to its previous precision level (Case 3);

The procedure of the proposed watermarking algorithm is summarized in Table I, including the embedding and recovery processes. The additional operations required in Cases 2 and 3 have been marked, respectively. From the discussions, it can be seen that our algorithm is quite suitable for the data obtained in the measurement.

IV. Application in the Measurement

To illustrate how to apply the proposed algorithm in the measurement, we take the 3-D geometry with coordinates for instance, including polygonal meshes and point clouds. A polygonal mesh approximates the surface of a 3-D object by specifying a set of vertices. In addition to the coordinates of vertices in 3-D space $\Re^3$, the connectivity between them is also contained in a polygonal mesh. In contrast, only the coordinates of the sampled points are recorded in a point cloud, which is obtained by 3-D scanning.

With the dissemination of 3-D models, such as the VRML models for representing 3-D graphics on the web, how to verify the authenticity and integrity of 3-D geometry has become an important issue. In the literature, quite a few watermarking algorithms (e.g., [24]–[28]) have been proposed to embed a tamperproof watermark in a 3-D mesh model. Since the geometry has been changed by the embedding process, it is desirable to recover the original object after extracting the embedded watermark. As the coordinates are represented by real numbers, the proposed algorithm can directly be applied to them.

Suppose that there are N vertices or points $\mathbf{P} = \{\mathbf{p_1}, \mathbf{p_2}, \ldots, \mathbf{p_N}\}$ in a 3-D geometry, where $\mathbf{p_i} = \{p_{ix}, p_{iy}, p_{iz}\}$ is the coordinates of a vertex or point. The watermarking algorithm can be implemented on the three sets of coordinates $\{p_{1x}, p_{2x}, \ldots, p_{Nx}\}$, $\{p_{1y}, p_{2y}, \ldots, p_{Ny}\}$, and $\{p_{1z}, p_{2z}, \ldots, p_{Nz}\}$ with the same quantization step size Δ and parameter a following the procedure in Section II-A, respectively. A watermark consisting of three $3N$ bits, as denoted by $\mathbf{W} = \{w_1, w_2, \ldots, w_{3N}\}$, can be embedded by modulating $\mathbf{P}$ to $\mathbf{G}' = \{\mathbf{g}'_1, \mathbf{g}'_2, \ldots, \mathbf{g}'_N\}$. A PR sequence $\mathbf{D} = \{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_N\}$ distributed in $(-(\Delta/2), (\Delta/2))$ is generated and added to $\mathbf{G}'$ to generate the watermarked geometry $\mathbf{G} = \{\mathbf{g_1}, \mathbf{g_2}, \ldots, \mathbf{g_N}\}$.

To extract the embedded data from $\mathbf{G}$, the PR sequence $\mathbf{D} = \{\mathbf{d_1}, \mathbf{d_2}, \ldots, \mathbf{d_N}\}$ should be subtracted to obtain the set of vectors $\mathbf{G}' = \{\mathbf{g}'_1, \mathbf{g}'_2, \ldots, \mathbf{g_N}'\}$. With the quantization step size Δ used in the embedding process, three strings of bit values can be retrieved from $\{g'_{1x}, g'_{2x}, \ldots, g'_{Nx}\}$, $\{g'_{1y}, g'_{2y}, \ldots, g'_{Ny}\}$, and $\{g'_{1z}, g'_{2z}, \ldots, g'_{Nz}\}$, respectively. In addition, the original coordinates $\{p_{1x}, p_{2x}, \ldots, p_{Nx}\}$, $\{p_{1y}, p_{2y}, \ldots, p_{Ny}\}$, and $\{p_{1z}, p_{2z}, \ldots, p_{Nz}\}$ can be recovered from $\mathbf{G}'$ with the quantization step size Δ and the parameter a, as shown in Section II-B. If the coordinates in the watermarked geometry can losslessly be compressed to make room for the error vector (i.e., Case 2 in Section III) or the precision of the watermarked geometry can be increased without changing the file size (i.e., Case 3 in Section III), then the recovered geometry will be identical to the original geometry. If neither the watermarked geometry is losslessly compressed nor its precision is increased, i.e., Case 1 in Section III, then the difference between the recovered and the original coordinates belongs to $\{-10^{-n}, 0, 10^{-n}\}$ by setting the parameter a within (2,3), where 10^{-n} is the precision level of coordinates in the original geometry. To make the embedded watermark invariant to the roundoff error, the quantization step size Δ should be larger than $(a/(a-2)) \times 10^{-n}$, as shown in (6). To verify the authenticity and integrity, the recovered geometry is regarded as intact only when the extracted watermark matches with the embedded watermark.

In industrial applications, the embedded watermark can include the content information (e.g., a hash value) and the metadata, such as the attributes and documentation of the original object, the copyright information, etc. In addition to 3-D coordinates, the watermarking algorithm can be applied to other measurement results. For every value in the measurement result, one bit can be embedded with the same quantization step size so that the embedded watermark can blindly be extracted. To exactly restore the original values, a convenient way to enable the exact recovery is to increase the precision of the watermarked data, as discussed in Case 3 of Section III. Even if the precision cannot be increased, only the smallest error may be caused in the last digit of each value.

TABLE II
VRML MODELS USED IN THE EXPERIMENTS

3D VRML models	Vertices	Polygons	Capacity (bits)	Bytes saved by lossless compression
lamp	676	1288	2028	0
pear	891	1704	2673	0
sgilogo	1224	1620	3672	0
pavilion	7334	5338	22002	19760
indigo	8389	10187	25167	16651
gears	24546	8182	73638	96247

V. PERFORMANCE ANALYSIS

We implemented the proposed algorithm on 3-D models in VRML format, as listed in Table II,[1] where the capacity of every model is also given. A secret key K was used as the seed of the PR number generator to generate the sequence to be added. The watermark can be some metadata of the original model and/or a hash value generated from the recoverable content. We tested the three cases listed in Section III, respectively. In Case 1, the precision of the watermarked geometry was the same as that of the original geometry. In Case 2, the redundancy in the watermarked geometry was exploited by losslessly compressing the floating-point numbers to make room for the error vector. However, only the models of "pavilion", "indigo", and "gears" consist of enough coordinates to make use of this method (see the number of bytes in Table II that can be saved by the lossless compression software WinRAR [29]). In Case 3, the coordinates in the watermarked models were increased from the precision level of 10^{-5} to 10^{-6}. When the coordinates in the original VRML models were represented by floating point numbers with a single precision, all of the watermarked models listed in Table II could still be stored after increasing the precision.

In the experiments, we assigned 2.1 to the parameter a. When the watermarked geometry was at the precision level of 10^{-6}, the quantization step size Δ should be greater than $(2.1\sqrt{3}/(2.1-2)) \times 10^{-6} \simeq 3.64 \times 10^{-5}$ to ensure that the embedded data can correctly be extracted. In case that the precision of the watermarked geometry cannot be increased and at the level of 10^{-5}, the quantization step size Δ should be greater than $(2.1\sqrt{3}/(2.1-2)) \times 10^{-5} \simeq 3.64 \times 10^{-4}$. The pictures rendered from the original, watermarked, and recovered VRML models of "sgilogo" and "gears" in Case 3 are shown in Fig. 3.

A. Imperceptibility and Reversibility

To represent the geometrical distortion of 3-D geometry, the 3-D SNR defined in [21] is used. The impact of watermark embedding could be tuned by the quantization step size Δ. If 0.00008 and 0.0005 were assigned to Δ in Case 3, then the obtained 3-D SNRs of the watermarked VRML model "indigo" were about 73.20 and 57.33 dB, respectively. In Case 2, the distortion of the watermarked geometry was aggravated by

[1]Downloaded from http://www.martinreddy.net/ukvrsig/vrml.html

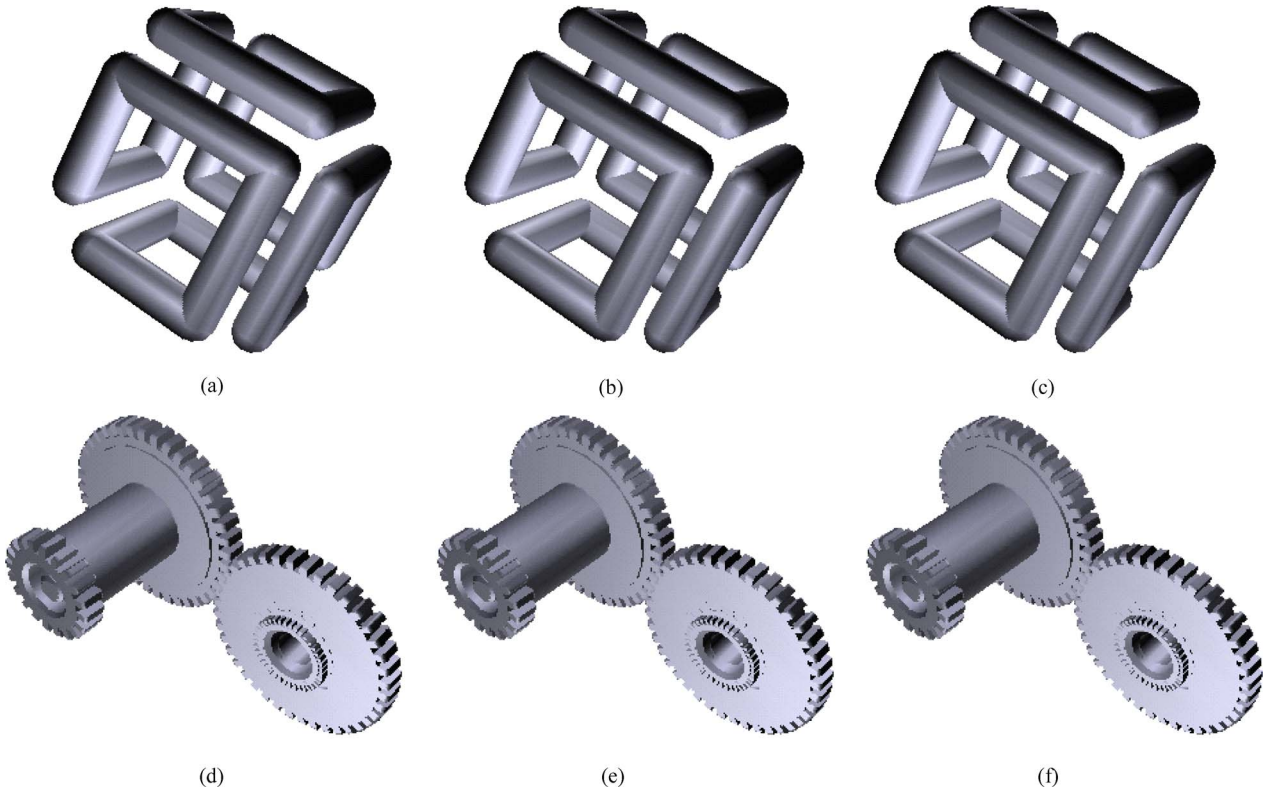

Fig. 3. (a) Original VRML model of "sgilogo". (b) The watermarked model of "sgilogo" generated by setting $\Delta = 0.00008$ and $a = 2.1$, as discussed in Case 3 of Section III. (c) Recovered model of "sgilogo". (d) Original VRML model of "gears". (e) Watermarked model of "gears". (f) Recovered model of "gears".

replacing the two least significant bytes in the mantissa part of every floating point number with the compressed bit string and the error vector. The 3-D SNRs of the watermarked models "pavilion," "indigo," and "gears" after the replacement were 44.03, 47.07, and 47.53 dB, respectively.

The 3-D SNR of the recovered geometry can be calculated in the same way. When the precision of the watermarked VRML model was increased from 10^{-5} to 10^{-6} as in Case 3, the recovered model was identical to the original model because the obtained 3-D SNR was *infinite* by setting Δ and the parameter a at 0.00008 and 2.1, respectively. The exact recovery could also be achieved in Case 2, but only for the large VRML models, such as "pavilion," "indigo," and "gears" in Table II.

In Case 1, neither the precision of the watermarked geometry is increased, nor the coordinates are compressed to save the error vector. Therefore, the original geometry could only be approximately recovered, and the 3-D SNRs of the recovered geometries were calculated. When 0.001 and 0.0005 were assigned to Δ, the 3-D SNR values of the recovered model "indigo" were both about 83.72 dB. As shown in Fig. 4, the 3-D SNR of the recovered geometry is always much higher than the 3-D SNR of the watermarked geometry, which is decreased when the quantization step size Δ is increased. It can be seen that the roundoff error is constant and small.

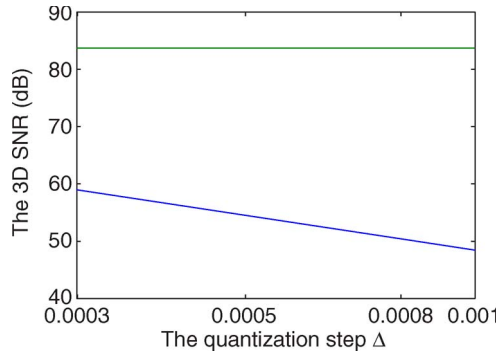

Fig. 4. By setting the parameter a at 2.1 in Case 1 in Section III, the 3-D SNR of the watermarked VRML model "indigo" decreases over the quantization step size Δ, as shown by the lower curve. In contrast, the 3-D SNR of the approximately recovered model remains constant, as shown by the upper curve.

B. Capacity

Since a position vector has three coordinates, three bit values can be embedded in it. Given N vertices in a polygonal mesh or N points in a point cloud, the data-hiding capacity of the implementation is $3N$ bits, which is higher than the previous works, such as $3N - 2$ bits in [26] and $N - 1$ bits in [21]. Different from the algorithm in [26] that is applicable to manifold triangle meshes, the capacity of our algorithm can be maximized for any 3-D geometry consisting of coordinates.

C. Authentication and Integrity Verification

The authenticity and integrity of a 3-D geometry can be verified by embedding a watermark into every coordinate and then comparing the extracted watermark with the embedded watermark. By assigning 0.00008 and 2.1 to the quantization step size Δ and the parameter a in Case 3, a watermark independent from the original geometry was embedded. The watermarked geometry underwent the following modifications: modifying

TABLE III
WATERMARKED GEOMETRY IS GENERATED BY ASSIGNING 0.00008 AND 2.1 TO THE PARAMETERS Δ AND a, AS DISCUSSED IN CASE 3 IN SECTION III. AFTER THE WATERMARKED GEOMETRY HAS BEEN PROCESSED BY THE FOLLOWING MODIFICATIONS, THE EXTRACTED WATERMARKS ARE COMPARED WITH THE ORIGINAL WATERMARKS TO OBTAIN THE VALUE OF H DEFINED IN (11)

3D VRML models	Modifying one vertex position	Moving two vertices oppositely	Adding Gaussian noise	Translation	Uniformly scaling	Rotation	Without subtracting the added sequence
lamp	0.9990	0.9980	0.8096	0.4117	0.4872	0.5039	0.7687
pear	0.9993	0.9988	0.8178	0.3793	0.4976	0.4822	0.7583
sgilogo	0.9995	0.9991	0.8153	0.4251	0.4989	0.5106	0.7549
pavilion	0.9999	0.9998	0.8151	0.4193	0.5016	0.5017	0.7695
indigo	0.9999	0.9999	0.8147	0.5655	0.5004	0.4976	0.7695
gears	0.9999	0.9999	0.8143	0.3856	0.4986	0.4964	0.7648

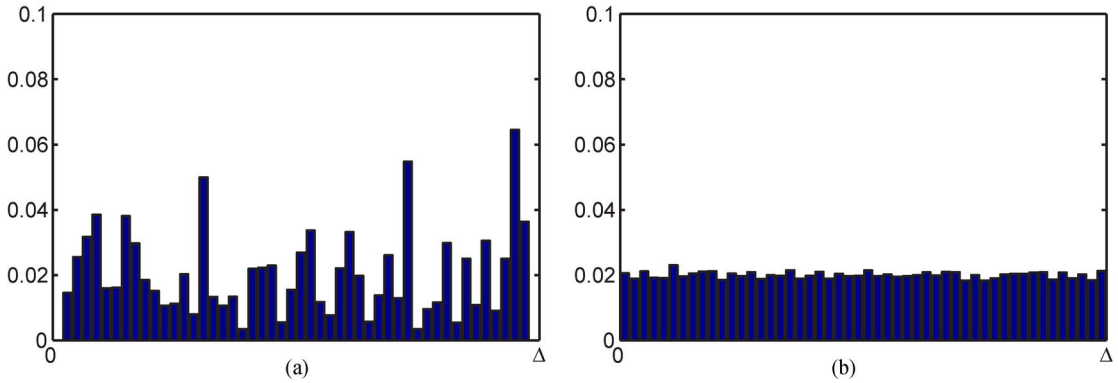

Fig. 5. PMFs of $(\Delta/2)+e_i$ and $((\Delta/2)+e_i+V_i)\%\Delta$ for the coordinates in the watermarked VRML model of "pavilion," with $a=2.1$ and $\Delta=0.037$. (a) PMF of $(\Delta/2)+e_i$. (b) PMF of $((\Delta/2)+e_i+V_i)\%\Delta$.

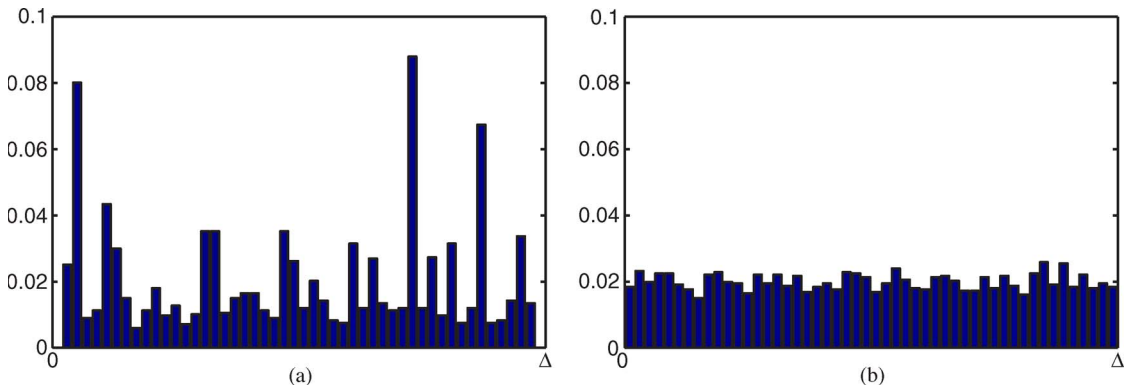

Fig. 6. PMFs of $(\Delta/2)+e_i$ and $((\Delta/2)+e_i+V_i)\%\Delta$ for the coordinates in the watermarked VRML model of "pear," with $a=2.1$ and $\Delta=0.0007$. (a) PMF of $(\Delta/2)+e_i$. (b) PMF of $((\Delta/2)+e_i+V_i)\%\Delta$.

one vertex position by adding the vector $\{4\Delta, 5\Delta, 6\Delta\}$, oppositely moving two vertices by adding the vector $\{2\Delta, 2\Delta, 2\Delta\}$ and $\{-2\Delta, -2\Delta, -2\Delta\}$, applying random permutations to every coordinate in the watermarked geometry according to Gaussian noise with zero mean and a variance of $\Delta^2/16$, translation, uniformly scaling, and rotation.

A string of bit values were extracted from the modified geometry, as denoted by $\mathbf{W}'=\{w'_1, w'_2, \ldots, w'_{3N}\}$, after performing the modifications, respectively. The extraction was also performed without subtracting the added sequence from the watermarked geometry. The extracted watermark $\mathbf{W}'$ was compared with the original watermark $\mathbf{W}=\{w_1, w_2, \ldots, w_{3N}\}$ by using the normalized Hamming distance

$$H=\frac{1}{3N}\sum_{I=1}^{3N} C\left(w'_i, w_i\right) \tag{11}$$

where $C(w'_i, w_i)$ is equal to 1 if $w'_i=w_i$ and equal to 0 otherwise. By comparing $\mathbf{W}'$ with $\mathbf{W}$ using (11), the value of H was calculated and listed in Table III. When slight modifications were made, the value of H was less than but close to 1. When the modifications were global and intense, the extracted values were greatly different from the embedded ones. The value was close to 0.75 if the watermark extraction was performed without subtracting the added sequence, which is distributed within $(-(\Delta/2), (\Delta/2))$. It can be seen that the embedded watermark was sensitive to all the modifications so that it is suitable for *strict* authentication. With a watermark independent from the original geometry, the geometrical modifications can be localized. If a hash value generated from the recoverable content is embedded instead, then the modification can easily be detected but cannot be localized to the tampered vertices or points.

D. Security Enhancement

By generating a PR sequence with a secret key and adding it to the watermarked object, the PMF of the remainders of the resulting values modulo the quantization step size is close to uniform distribution. In addition to Figs. 1 and 2, the PMFs of $(\Delta/2)+e_i$ and $((\Delta/2)+e_i+V_i)\%\Delta$ for the coordinates in the watermarked VRML models of "pavilion" and "pear" were calculated in the same way, as shown in Figs. 5 and 6, respectively. The similar results were obtained for the other VRML models in Table II. Since there is no gap in the PMF of the remainders after adding the PR sequence, it becomes much harder to estimate the quantizer employed in the modulation. Even if the quantization step size is known, the embedded watermark cannot correctly be extracted without subtracting the added sequence, as shown in Table III. As the PR sequence is mixed with the modulation information, it is hard to subtract it from the watermarked object without the secret key. Therefore, one cannot illegally extract the embedded watermark and recover the original object. Hence, the security of the watermarking algorithm has been enhanced by the adoption of a PR sequence.

VI. Conclusion

In this paper, a watermarking algorithm has been proposed for the objects represented by floating- or fixed-point numbers. By keeping the modulation information in the watermarked object, the original object can approximately be recovered, whereas exact recovery is possible by further exploiting the redundancy. The security of the algorithm has been analyzed and enhanced by adding a PR sequence to the watermarked object. In the recovery process, the added sequence should be subtracted to extract the watermark and recover the original object.

We have discussed the applicability of the algorithm. It has been shown that it is quite suitable for the data obtained in the measurement. Moreover, it has been applied to 3-D objects consisting of coordinates to embed some useful information. The implementation on 3-D VRML models has shown the efficacy of our algorithm.

References

[1] I. J. Cox, J. Killian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.

[2] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inf. Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.

[3] J. J. Eggers, R. Bauml, R. Tzschoppe, and B. Girod, "Scalar Costa scheme for information embedding," *IEEE Trans. Signal Process.*, vol. 51, no. 4, pp. 1003–1019, Apr. 2003.

[4] M. A. Suhail and M. S. Obaidat, "Digital watermarking-based DCT and JPEG model," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 5, pp. 1640–1647, Oct. 2003.

[5] S. Biswas, S. R. Das, and E. M. Petriu, "An adaptive compressed MPEG-2 video watermarking scheme," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 5, pp. 1853–1861, Oct. 2005.

[6] L. Cai, R. Tu, J. Zhao, and Y. Mao, "Speech quality evaluation: A new application of digital watermarking," *IEEE Trans. Instrum. Meas.*, vol. 56, no. 1, pp. 45–55, Feb. 2007.

[7] A. Menezes, P. van Oorchot, and S. Vanstone, *Handbook of Applied Cryptography*. Boca Raton, FL: CRC Press, 1997.

[8] C. W. Honsinger, P. Jones, M. Rabbani, and J. C. Stoffel, "Lossless recovery of an original image containing embedded data," U.S. Patent 6 791, Aug. 21, 2001.

[9] B. Macq and F. Deweyand, "Trusted headers for medical images," in *Proc. DFG VIII-DII Watermarking Workshop*, Erlangen, Germany, Oct. 1999, pp. 1–13.

[10] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication," in *Proc. SPIE, Security Watermarking Multimedia Contents*, San Jose, CA, Jan. 2001, vol. 4314, pp. 197–208.

[11] M. Celik, G. Sharma, M. Tekalp, and E. Saber, "Reversible data hiding," in *Proc. IEEE Int. Conf. Image Process.*, Rochester, NY, Sep. 2002, vol. 2, pp. 157–160.

[12] M. Goljan, J. Fridrich, and R. Du, "Distortion-free data embedding for images," in *Proc. 4th Inf. Hiding Workshop*, New York, Apr. 2001, vol. 2137, pp. 27–41.

[13] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 890–896, Aug. 2003.

[14] D. M. Thodi and J. J. Rodriguez, "Prediction-error based reversible watermarking," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2004, vol. 3, pp. 1549–1552.

[15] G. Xuan, J. Zhu, J. Chen, Y. Q. Shi, Z. Ni, and W. Su, "Distortionless data hiding based on integer wavelet transform," *Electron. Lett.*, vol. 38, no. 25, pp. 1646–1648, Dec. 2002.

[16] C. De Vleeschouwer, J. E. Delaigle, and B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *IEEE Trans. Multimedia*, vol. 5, no. 1, pp. 97–105, Mar. 2003.

[17] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 3, pp. 354–362, Mar. 2006.

[18] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk, "The digital Michelangelo project: 3D scanning of large statues," in *Proc. ACM SIGGRAPH*, Aug. 2000, pp. 131–144.

[19] *High Dynamic Range Imaging*. [Online]. Available: http://en.wikipedia.org/wiki/High_dynamic_range_imaging

[20] H. T. Wu and Y. M. Cheung, "A reversible data hiding approach to mesh authentication," in *Proc. IEEE/WIC/ACM Int. Conf. WI*, Sep. 2005, pp. 774–777.

[21] Y. M. Cheung and H. T. Wu, "A sequential quantization strategy for data embedding and integrity verification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 8, pp. 1007–1016, Aug. 2007.

[22] *The Virtual Reality Modeling Language*, ISO/IEC DIS 14772-1, 1997.

[23] *Standard for Binary Floating Point Arithmetic*, ANSI/IEEE Std. 754-1985, 1985.

[24] B. L. Yeo and M. M. Yeung, "Watermarking 3D objects for verification," *IEEE Comput. Graph. Appl.*, vol. 19, no. 1, pp. 36–45, Jan./Feb. 1999.

[25] O. Benedens and C. Busch, "Towards blind detection of robust watermarks in polygonal models," in *Proc. EUROGRAPHICS Comput. Graph. Forum*, Aug. 2000, vol. 19, pp. 199–208.

[26] F. Cayre, O. Devillers, F. Schmitt, and H. Maitre, "Watermarking 3D triangle meshes for authentication and integrity," INRIA, Rennes, France, INRIA Research Rep. RR-5223, Jun. 2004.

[27] H. Y. S. Lin, H. Y. M. Liao, C. S. Lu, and J. C. Lin, "Fragile watermarking for authenticating 3-D polygonal meshes," *IEEE Trans. Multimedia*, vol. 7, no. 6, pp. 997–1006, Dec. 2005.

[28] H. T. Wu and Y. M. Cheung, "Public Authentication of 3-D Mesh Models," in *Proc. IEEE/WIC/ACM Int. Conf. WI*, Hong Kong, Dec. 2006, pp. 940–948.

[29] *The WinRAR Software*. [Online]. Available: http://www.rarlab.com/

Hao-Tian Wu, photograph and biography not available at the time of publication.

Yiu-Ming Cheung (S'96–M'00–SM'06), photograph and biography not available at the time of publication.