Fragile Watermarking for Authenticating 3-D Polygonal Meshes

Hsueh-Yi Sean Lin, Hong-Yuan Mark Liao, *Senior Member, IEEE*, Chun-Shien Lu, *Member, IEEE*, and Ja-Chen Lin

***Abstract*—Designing a powerful fragile watermarking technique for authenticating three-dimensional (3-D) polygonal meshes is a very difficult task. Yeo and Yeung [34] were first to propose a fragile watermarking method to perform authentication of 3-D polygonal meshes. Although their method can authenticate the integrity of 3-D polygonal meshes, it cannot be used for localization of changes. In addition, it is unable to distinguish malicious attacks from incidental data processings. In this paper, we trade off the causality problem in Yeo and Yeung's method for a new fragile watermarking scheme. The proposed scheme can not only achieve localization of malicious modifications in visual inspection, but also is immune to certain incidental data processings (such as quantization of vertex coordinates and vertex reordering). During the process of watermark embedding, a local mesh parameterization approach is employed to perturb the coordinates of invalid vertices while cautiously maintaining the visual appearance of the original model. Since the proposed embedding method is independent of the order of vertices, the hidden watermark is immune to some attacks, such as vertex reordering. In addition, the proposed method can be used to perform region-based tampering detection. The experimental results have shown that the proposed fragile watermarking scheme is indeed powerful.**

***Index Terms*—Authentication, fragile watermarking, parameterization, polygonal meshes, tampering detection.**

I. Introduction

TRANSFERRING digitized media via the Internet has become very popular in recent years. Content providers who present or sell their products through networks are, however, faced with the copyright protection problem. In order to properly protect the rights of a content owner, it is desirable to develop a robust protection scheme that can prevent digital contents from being stolen or illegally distributed. From a user's point of view, after receiving a piece of digital content, he/she usually needs to verify the integrity of the content. As a result, there should be an authentication mechanism that can be used to perform the verification task. With the rapid advance of watermarking technologies in recent years, many investigators have devoted themselves to conducting research in this fast growing area. According to the objectives that a watermarking technique may achieve, two main-stream digital watermarking categories are: robust watermarking and fragile watermarking. While the former aims to achieve intellectual property protection of digital contents, the latter attempts to authenticate the integrity of digital contents.

There are a great number of existing robust watermarking algorithms designed to protect three-dimensional (3-D) graphic models [1]–[3], [6], [7], [16], [18], [21]–[27], [33], [36]. Their common purpose is to provide a robust way to protect target contents when attacks are encountered. The existing fragile watermarking algorithms that are designed to authenticate 3-D graphic models are relatively few. In [10], Fornaro and Sanna proposed a public key approach to authenticating constructuve solid geometry (CSG) models. In [17], Kankanhalli *et al.*, proposed the use of content-based signature to authenticate 3-D volume data. In [34], Yeo and Yeung proposed a fragile watermarking algorithm for authenticating 3-D polygonal meshes. They embed a fragile watermark by iteratively perturbing vertex coordinates until a predefined hash function applied to each vertex matches the other predefined hash function applied to that vertex. Since their embedding algorithm relies heavily on an ordered traversal of vertices, it is capable of detecting object cropping. However, the consideration of causality disables it from localization of changes and robustness against vertex reordering. In addition, particular attacks, such as floating-point truncation or quantization, applied to vertex coordinates might increase the false-alarm probability of tampering detection.

In this paper, we trade off the causality problem in Yeo and Yeung's method for a new fragile watermarking scheme. The proposed scheme can not only achieve localization of malicious modifications in visual inspection, but also is immune to the aforementioned unintentional data processings. In addition, the allowable range for alternating a vertex is explicitly defined so that the new scheme is able to tolerate quantization of vertex coordinates (up to a certain amount). During the process of watermark embedding, a local mesh parameterization approach is employed to perturb the coordinates of invalid vertices while cautiously maintaining the visual appearance of the original model. Since the proposed embedding method is independent of the order of vertices, the hidden watermark is immune to some vertex order-dependent attacks, such as vertex reordering.

The remainder of this paper is organized as follows. In Section II, Yeo and Yeung's scheme for authenticating 3-D polygonal meshes is briefly reviewed. In Section III, the proposed fragile watermarking method is described in detail. Experimental results are given in Section IV. Finally, conclusions are drawn in Section V.

Manuscript received June 17, 2003; revised September 25, 2004. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ton A. C. M. Kalker.

H.-Y. S. Lin and J.-C. Lin are with the Department of Computer and Information Science, National Chiao-Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: hylin@cis.nctu.edu.tw; jclin@cis.nctu.edu.tw).

H.-Y. M. Liao and C.-S Lu are with the Institute of Information Science, Academia Sinica, Taipei, Taiwan 115, R.O.C. (e-mail: liao@iis.sinica.edu.tw; lcs@iis.sinica.edu.tw).

Digital Object Identifier 10.1109/TMM.2005.858412

II. Yeo and Yeung's Approach and Its Drawbacks

In [34], Yeo and Yeung proposed a novel fragile watermarking algorithm which can be applied to authenticate 3-D polygonal meshes. In Yeo and Yeung's scheme [34], there are three major components, i.e., two predefined hash functions and an embedding process. For a given vertex, the vertex is identified as valid if and only if the values calculated by both hash functions are identical. Otherwise, the vertex is identified as invalid. During the authentication process, invalid vertices are considered as the set of vertices that has been tampered with. On the other hand, valid vertices indicate the set of vertices which has never been modified. In the embedding process, the coordinates of valid vertices are kept unchanged, but those of invalid vertices are iteratively perturbed until each of them becomes valid.

The first step in Yeo and Yeung's approach is to compute location indices. In this step, the first hash function is defined by a conversion function and associated with a given watermark pattern WM. The conversion function is used to convert a vertex coordinate $v = (v_x, v_y, v_z)$ into a location index $L = (L_x, L_y)$. The idea behind the conversion function is to map a 3-D coordinate onto a two-dimensional plane formed by a watermark pattern of dimension $WM\_X\_SIZE \times WM\_Y\_SIZE$. As a result, the location index L is used to point to a particular position in the watermark pattern. Then, the content of that particular position $WM(L)$ (either 0 or 1) is used for the purpose of comparison. Since the conversion function defined in [34] calculates the centroid of the neighboring vertices of a given vertex, the causality problem occurs. Furthermore, the traversal of vertices during the alternation of vertex coordinates must take causality into account so as to avoid error propagation.

The second step in Yeo and Yeung's approach is to compute value indices. In this step, the second hash function is related to a set of look-up tables (LUTs), i.e., K_1, K_2, and K_3. These LUTs, which are composed of sequences of bits, are generated and protected by an authentication key. Yeo and Yeung [34] proposed to convert each component of a vertex coordinate into an integer number so as to index it into each of the LUTs. The content of an indexed location is either 0 or 1. The three binary values derived from the three coordinates $p = (p_1, p_2, p_3)$ are then XOR processed to generate a final binary value. This binary value $K(p)$ is used as one of the components for deciding whether the current vertex is valid or not. If the vertex is not valid, then it is perturbed until it is valid. The amount of change that makes this vertex valid is the watermark embedded.

After establishing the above-mentioned two hash functions, the next step is to perturb the coordinates of all invalid vertices until they become valid. In [34], the authors proposed an iterative procedure which can gradually perturb an invalid vertex until both hash functions are matched. On the one hand, in order to maintain transparency, the embedding procedure must traverse in an orderly manner each vertex during the alteration of vertex coordinates. In addition, the ordering of vertices must be maintained during the watermark extraction process. The benefit of taking the causality into account is for protection against changes of connectivity (in particular cropping). However, the drawback is that their method cannot achieve localization of malicious modifications in visual inspection. In addition, their method cannot tolerate certain incidental modifications, such as quantization of vertex coordinates and vertex reordering. This drawback to some extent limits the power of Yeo and Yeung's method. In this paper, we shall propose a new scheme that is more powerful than the existing fragile watermarking algorithms.

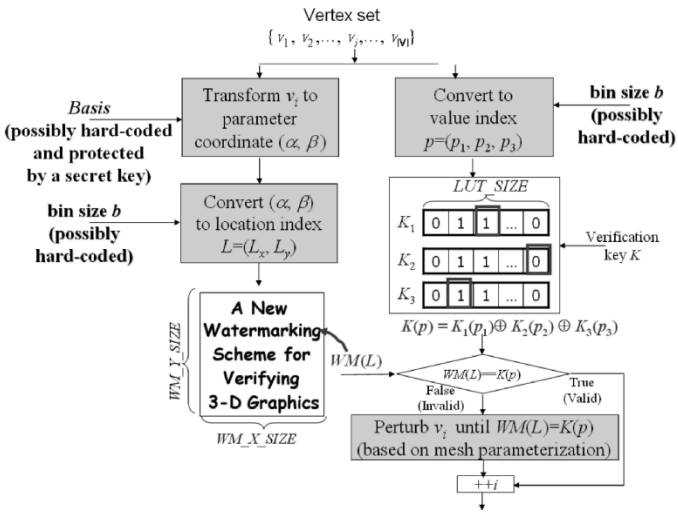

Fig. 1. Flowchart of the proposed authentication scheme for 3-D polygonal meshes.

III. Proposed Fragile Watermarking Method

In this section, we shall propose a new fragile watermarking scheme for authenticating 3-D polygonal meshes. In order to tackle the issues that were not handled by Yeo and Yeung [34], we employ the following concepts. 1) Each hash function can be designed so as to form a binary state space particularly helpful for defining the domain of allowable alternation for a given vertex. Accordingly, the domain of acceptable alternation for a given vertex can be defined as the intersection of the binary state spaces where the values of both hash functions match each other. 2) In order to resolve the causality problem, the conversion function used in the first hash function can be designed to simply perform the mapping from the 3-D space to a 2-D plane without considering the neighboring vertices of a vertex. Based on the above two concepts, we have designed a new scheme, which is shown in Fig. 1. With the new authentication scheme, malicious attacks applied to 3-D polygonal meshes can be easily distinguished from certain incidental modifications. In what follows, we shall describe our authentication scheme in more detail.

A. Computing Location Indices

Since the conversion function used in the first hash function (the left hand side of Fig. 1) aims to calculate the location index that can be used to locate a particular bit in the watermark pattern, any functions that can transform a 3-D coordinate into a 2-D coordinate can serve this purpose. Therefore, it is possible to use some parameterization schemes to achieve the goal. As mentioned in the previous section, Yeo and Yeung did not use an analytical method to perturb invalid vertices. However,

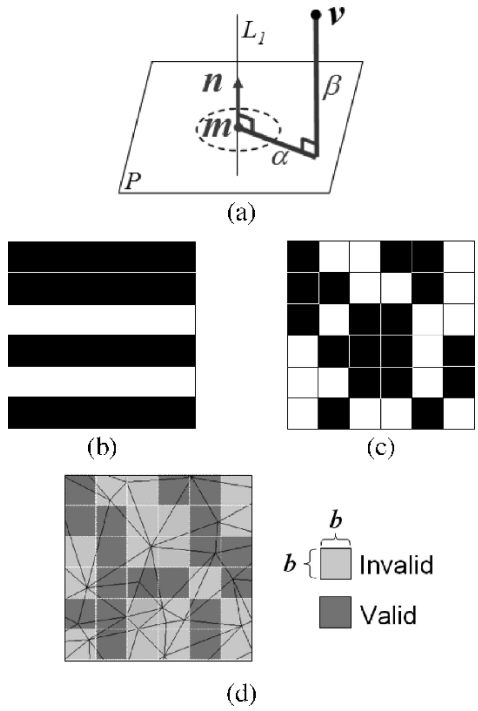

Fig. 2. Illustration of the robustness construction: (a) basis for cylindrical parameterization [14]; (b) side view of the binary state space formed by the quantized cylindrical parameterization domain; (c) the side view of the binary state space formed by the conversion function for computing value indices; and (d) the two binary state spaces superimposed on a sidepiece of the cylindrical mesh with irregular connectivity.

a systematic perturbation strategy is always preferable. Therefore, we propose to adopt the parameterization-based approach to make the vertex perturbation process analytic. For the purpose of clarity, we propose to split the location index computation process into two steps.

Step 1: Given a vertex coordinate v, the specified parameterization $S : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ converts the vertex coordinate into a parameter coordinate. We propose to use so-called cylindrical parameterization [14] to perform the conversion task. The procedure involved in performing cylindrical parameterization is as follows [14]:

Given an oriented 3-D point, it is composed of a 3-D point m and its orientation n. As shown in Fig. 2(a), a cylindrical parameterization process[1] can be expressed as

$$S_{m,n}(v) \rightarrow (\alpha, \beta) = \left(\sqrt{||v-m||^2 - (n \cdot (v-m))^2}, n \cdot (v-m) \right) \quad (1)$$

where (α, β) is the coordinate in the parameter domain. The range for each dimension of the parameter domain is $\alpha \in [0, \infty)$ and $\beta \in (-\infty, \infty)$, respectively.

[1]Note that although an oriented point defines five degrees of freedom (DOF) basis (m, n), the proposed method is not immune to geometrical transformations. This results from the fact that whether a vertex is valid or not is guarded by the two hash functions.

Step 2: Convert the parameter coordinate formed in Step 1 into the so-called bin coordinate, i.e., the location index (L_x, L_y). This conversion can be accomplished by quantizing the parameter domain. In addition, a modulus operator is required to map them onto the dimension of a watermark pattern. In what follows, we shall describe how the parameter domains are quantized. Assume that the size of a 2-dimensional watermark pattern is $WM\_X\_SIZE \times WM\_Y\_SIZE$, the quantization formula for a cylindrical parameterization domain is as follows:

$$L = (L_x, L_y) = \left(\left\lfloor \frac{\alpha}{b} \right\rfloor \% WM\_X\_SIZE, \left\lfloor \frac{\beta}{b} \right\rfloor \% WM\_Y\_SIZE \right) \quad (2)$$

where b is the quantization step for ordinary numeric values and % represents a modulus operator.

A very important feature of the above design is that the quantized parameterization domain and the watermark pattern together form a binary state space. Such a state space is helpful for defining a legal domain of alternation for a given vertex. The side-view of the binary state space corresponding to the quantized cylindrical parameterization domain is illustrated in Fig. 2(b).

B. Computing Value Indices

Even though any functions for converting a floating-point number into an integer can be used to calculate value indices, the following conversion function was designed since it is able to form a binary state space. Assuming that the size of each LUT is LUT_SIZE, the conversion function is formulated as

$$p = (p_1, p_2, p_3) = \left(\left\lfloor \frac{v_x}{b} \right\rfloor \% LUT\_SIZE, \left\lfloor \frac{v_y}{b} \right\rfloor \% LUT\_SIZE, \left\lfloor \frac{v_z}{b} \right\rfloor \% LUT\_SIZE \right) \quad (3)$$

where b is the same quantization step as used to compute location indices.

The side-view of the binary state space corresponding to the above conversion function is illustrated in Fig. 2(c). In addition, Fig. 2(d) reveals that the domain of acceptable alternation for a given vertex can be defined as the intersection of the binary state spaces where the values of both hash functions applied to that vertex match each other. More precisely, for a valid vertex the displacement applied to its original coordinates will depend on the value of (α, β) and thus it will make (L_x, L_y) change as well. As long as the displacement for both location and value indices does not vary beyond the aforementioned domain of acceptable alternation, the vertex will be identified as intact by our scheme. As a result, the encoded location and value indices will be robust to a certain extent of quantization.

C. Watermark Embedding

Since both hash functions have been well-designed to define the domain of acceptable alternation for a given vertex, the

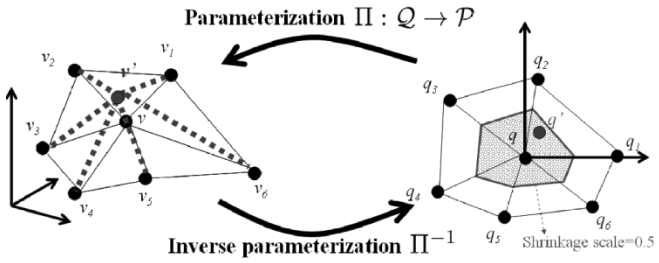

Fig. 3. Proposed alternation procedure for an invalid vertex.

embedding procedure can focus on perturbing the coordinates of invalid vertices while maintaining transparency. In the remeshing-related literatures [9], [20], [28], [31], [37], the points over the surface of the polygonal model have frequently been used for resampling the geometry of a model. We, therefore, apply a local mesh parameterization approach proposed in [20] for finding a valid point on the surface of a polygonal mesh. Assume that the polygonal model to be watermarked is a closed and oriented two-manifold mesh that has been triangulated, our method is as follows: Given an invalid vertex $v \in \mathbb{R}^3$ and its neighboring vertices in the counter-clockwise order $v_1, v_2, \ldots, v_{|N(v)|} \in \mathbb{R}^3$, where $|N(v)|$ is the number of v's neighboring vertices, the proposed alternation procedure for an invalid vertex is divided into five steps, which can be explained with the help of Fig. 3. The details of the five steps are as follows.

Step 1: Transform the vertex coordinate v into the parameter coordinate $q \in \mathbb{R}^2$ and its neighboring vertices $v_1, v_2, \ldots, v_{|N(v)|}$ to $q_1, q_2, \ldots, q_{|N(v)|} \in \mathbb{R}^2$, respectively, using arc-length parameterization. Let $ang(a,b,c)$ be the angle formed by vectors $\overrightarrow{ba}$ and $\overrightarrow{bc}$. Then, the parameter coordinates are provided with the following properties [9]:

$$\|q_k - q\| = \|v_k - v\| \tag{4}$$

$$ang(q_k, q, q_{k+1}) = 2\pi \cdot \frac{ang(v_k, v, v_{k+1})}{\theta} \tag{5}$$

where

$$\theta = \sum_{k=1}^{|N(v)|} ang(v_k, v, v_{k+1}),\ v_{|N(v)|+1} = v_1$$

$$q_{|N(v)|+1} = q_1, \quad \text{and} \quad k = 1, \ldots, |N(v)|.$$

If we set $q = (0,0)$ and $q_1 = (\|v_1 - v\|, 0)$, the parameter coordinates $q_2, q_3, \ldots, q_{|N(v)|}$ can be easily derived from (4) and (5). Hence, $q_1, q_2, \ldots, q_{|N(v)|}$ form the boundary vertices of the star-shaped planar polygon $\mathcal{Q}$ with q in its kernel. In addition, $v_1, v_2, \ldots, v_{|N(v)|}$ are the boundary vertices of the polygon $\mathcal{P}$ with one internal vertex v. Let $\sqcup_k$ denote the triangle formed by the parameter coordinates q, q_k, q_{k+1} and $\mathcal{T}_k$ denote the triangle formed by the vertex coordinates v, v_k, v_{k+1} for $k = 1, \ldots, |N(v)|$. Then, the two triangle sets $\{\sqcup_k\}$ and $\{\mathcal{T}_k\}$ form the triangulation of the planar polygon $\mathcal{Q}$ and the polygon $\mathcal{P}$, respectively.

Step 2: Establish the local mesh parameterization $\Pi : \mathcal{Q} \to \mathcal{P}$ by means of the well-known barycentric mapping. Let $\hat{q}$ denote an arbitrary point inside the planar polygon $\mathcal{Q}$ and $area(a,b,c)$ denote the signed area of the triangle formed by the vertices a, b, c. Then, there exists a unique $t \in \{1, \ldots, |N(v)|\}$ such that the barycentric coordinates of $\hat{q}$ will correspond to the triangle $\sqcup_t$ and have the following forms:

$$\lambda_{t,1} = \frac{area(\hat{q}, q_t, q_{t+1})}{area(q, q_t, q_{t+1})}, \quad \lambda_{t,2} = \frac{area(q, \hat{q}, q_{t+1})}{area(q, q_t, q_{t+1})}$$

$$\lambda_{t,3} = \frac{area(q, q_t, \hat{q})}{area(q, q_t, q_{t+1})}. \tag{6}$$

The three barycentric coordinate components are all of the same sign. Hence, the corresponding point $\hat{v}$ on the surface of the polygon $\mathcal{P}$ can be represented as a combination of the points v, v_t, v_{t+1} with respect to $\mathcal{T}_t$ as follows:

$$\hat{v} = \lambda_{t,1} v + \lambda_{t,2} v_t + \lambda_{t,3} v_{t+1}. \tag{7}$$

Step 3: Define an allowable region for alternating an invalid vertex in the parameter domain. Let the region be a shrunken ring whose origin is the parameter coordinate, q, and let the scale for shrinkage be 0.5. (As shown in Fig. 2(d), for some invalid vertices to find a valid state may sacrifice a great deal of the original quality. As a result, the shrunken ring defined here can be regarded as the maximum bound of distortion induced by alternating an invalid vertex. In addition, it can avoid geometrical degeneracies, like triangle flipping, T-joints, etc.)

Step 4: Distribute a set of points $\tilde{\mathbf{q}} = \{\tilde{q}_i \in \mathbb{R}^2 : i = 1, \ldots, r\}$ randomly on the allowable region.[2] Next, find a new parameter coordinate $q' \in \tilde{\mathbf{q}}$ satisfying the condition

$$WM\left(L\left(S_{m,n}\left(\Pi(q')\right)\right)\right) = K\left(p\left(S_{m,n}\left(\Pi(q')\right)\right)\right) \tag{8}$$

where Π is the barycentric mapping derived from (6) and (7). If there does not exist such a new parameter coordinate, alternation for the current invalid vertex is skipped, and $q' = q$ is assigned.

Step 5: Record the new vertex coordinate $v' = \Pi(q')$.

Note that the set of random points generated in Step 4 can be sorted according to its geometric distance to the parameter coordinate q, in such a way that the new parameter coordinate q' can be chosen not only satisfying (8) but also minimizing the distortion. Currently, this feature has not been considered in our implementation since the maximum distortion has been bounded as described in Step 3. As for maximizing the robustness, the domain of acceptable alternation can be mapped onto the parameter domain using the inverse of the parameterization $\Pi^{-1} : \mathcal{P} \to \mathcal{Q}$. Then, the new parameter coordinate q' that maximizes the robustness can be determined. Since the efficiency of

[2]A random point in a triangle is generated using the method described in [32].

the algorithm would be degraded, this feature has not been implemented in our system. As for the performance of our method, it is a natural outcome of the Step 4 that a certain amount of invalid vertices may remain untouched/invalid. We, therefore, propose some possible solutions to optimize the performance in the following section.

D. Analysis and Discussion

In this section, we shall conduct a thorough analysis of our authentication scheme for 3-D polygonal meshes. The watermarking parameters that can influence the quality of transparency and robustness are the shrinkage scale and bin size. On the other hand, we also know that the correlation value C can never reach 1. Therefore, we shall examine several crucial issues: 1) how to optimize the performance so that C can be very close to 1; 2) how to balance the competition between transparency and capacity using the shrinkage scale; and 3) how to guarantee the robustness of a hidden watermark. Before discussing the above mentioned issues, we adopt the correlation value used by Yeo and Yeung [34] and formulated it as

$$C = \frac{|\{v : K(p(v)) = WM(L(v))\}|}{|V|} \tag{9}$$

where V is the vertex set of a mesh and $|V|$ is the total number of vertices. Note that the correlation C is the ratio of the number of valid vertices to the total number of vertices (instead of a linear correlation coefficient). In what follows, we shall discuss the aforementioned issues.

First of all, we aim to optimize the performance of our algorithm so that the watermark correlation value C can be very close to 1. In our investigation, there are two possible solutions to optimize the performance. The first solution is to adopt a smaller quantization step, which would increase the possibility of finding a valid state. Such an approach will be a great benefit to the maintenance of transparency. However, the drawback is that the robustness would be sacrificed as well. An alternative solution is to make the spacing between vertices regular while maintaining the shape of a 3-D mesh. In such an approach, the robustness can benefit greatly from the specified quantization step (i.e., the bin size). However, the drawback is that the shape of the mesh would be simplified significantly when the spacing between vertices is increased. Our intention here is to maintain the robustness when encountering certain incidental modifications, such as vertex quantization and noise addition. We, therefore, picked five different models to generate analysis models with different mesh resolutions using a mesh resolution control algorithm described in [15].[3] Furthermore, for each model, we generated five analysis models corresponding to different mesh resolutions. Thirty analysis models and their mesh resolutions are listed in Table I. Fig. 4 shows the flat-shaded HIV model and its analysis models corresponding to five different mesh resolutions. In the watermarking process, we fixed the shrinkage scale as 0.5 and the bin size as 2. With varied mesh resolution levels, our fragile watermark was embedded into each model to test the effect of the mesh resolution on the watermark correlation value. In addition, we ran each test five times using different keys and reported the median value. Fig. 5(a) shows the effect of different mesh resolutions on the watermark correlation value.

[3]In [15], the resolution of a mesh is defined as the median of its edge length histogram. In addition, the edge length spread is defined as the half-width (upper quartile minus lower quartile) of the histogram. The goal of the mesh resolution algorithm is to adjust the resolution of the original mesh to a desired resolution while minimizing the edge length spread of the histogram.

TABLE I
LIST OF 30 TRIANGULATED MESHES USED IN THE ANALYSIS

Model	Number of vertices/faces	Mesh resolution
spock	16386/32768	1.974926
spock-lv1	11543/23082	2.828352
spock-lv2	3604/7204	5.127024
spock-lv3	1819/3634	7.390215
spock-lv4	878/1752	10.350958
spock-lv5	426/848	14.553292
skull	20002/40000	1.524550
skull-lv1	9559/19114	2.876386
skull-lv2	3063/6122	5.138119
skull-lv3	1418/2832	7.652043
skull-lv4	747/1490	10.350958
skull-lv5	365/726	14.622301
holes3	5884/11776	3.581405
holes3-lv1	10620/21248	2.871731
holes3-lv2	3429/6866	5.061736
holes3-lv3	1555/3118	7.652731
holes3-lv4	847/1702	10.258323
holes3-lv5	410/828	14.520646
HIV	9988/20000	1.493200
HIV-lv1	2691/5398	2.915812
HIV-lv2	683/1372	5.233385
HIV-lv3	291/594	7.255211
HIV-lv4	135/270	10.562971
HIV-lv5	62/124	14.92876
isis	46912/93820	1.217783
isis-lv1	8727/17450	2.879494
isis-lv2	2765/5526	5.157736
isis-lv3	1347/2690	7.480489
isis-lv4	658/1312	10.380298
isis-lv5	329/654	14.454476

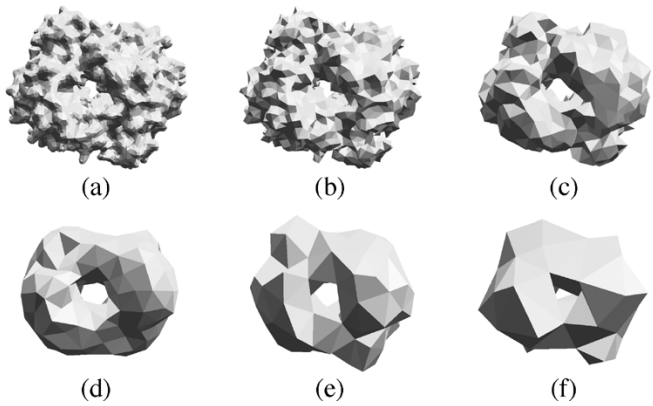

Fig. 4. Analysis models for the HIV protease surface model: (a) original HIV model; (b) HIV-lv1 model; (c) HIV-lv2 model; (d) HIV-lv3 model; (e) HIV-lv4 model; (f) HIV-lv5 model.

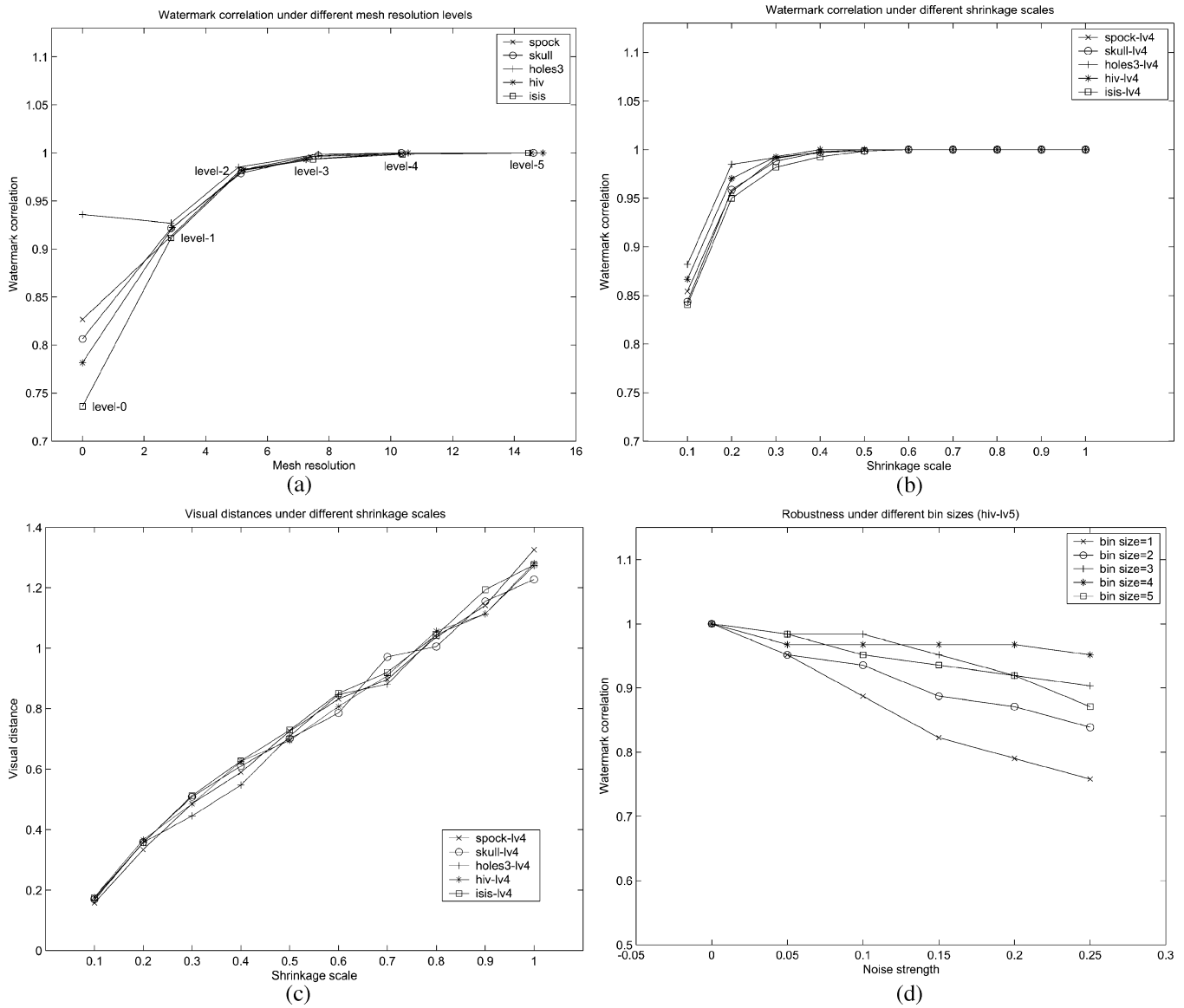

Fig. 5. (a) Effect of the mesh resolution on the watermark correlation value. Note that the mesh resolution of "0" indicates that the original models were not influenced by the mesh resolution control algorithm; (b) effect of the shrinkage scale on the watermark correlation value; (c) effect of the shrinkage scale on the transparency of our fragile watermark; and (d) robustness under different bin sizes for the HIV-lv5 model.

Obviously, the curves shown in Fig. 5(a) reveal that a polygonal mesh with higher mesh resolution would possess higher capacity for watermarking.

In order to investigate how the shrinkage scale can force a compromise between transparency and capacity, a suitable visual metric was needed to evaluate the difference between the original model M and the watermarked model M'. In [19], Karni and Gotsman proposed the use of Root-Mean-Square measure plus a Laplacian-based visual metric to capture human visual preceptibilities. The RMS metric simply captures the geometric distance between corresponding vertices in both models. On the other hand, the Laplacian-based metric can capture more subtle visual properties (such as smoothness) with respect to both the topology and geometry. The geometric Laplacian operator applied to a vertex v_i is defined as

$$GL(v_i) = v_i - \frac{\sum_{j \in n(i)} l_{ij}^{-1} v_j}{\sum_{j \in n(i)} l_{ij}^{-1}} \tag{10}$$

where $n(i)$ is the set of indices of v_i's neighboring vertices, and l_{ij} is the geometric distance between vertices i and j. Hence, the visual difference between the original model M and the watermarked model M' can be expressed as

$$diff(M, M') = \frac{1}{2|V|} \times \left(\sum_{i=1}^{|V|} ||v_i - v_i'|| + \sum_{i=1}^{|V|} ||GL(v_i) - GL(v_i')|| \right). \tag{11}$$

In the mesh-based watermarking literature [7], the above mentioned visual metric has been used to capture the geometric distortion between two models. We, therefore, adopted this visual metric to measure the transparency. In this analysis, we picked five models that were at the fourth resolution. We chose the bin size and the shrinkage scale as 2 and 0.5, respectively. With various shrinkage scales, our fragile watermark was embedded into each model for transparency and capacity tests. In the same way,

we ran each test five times using different keys and reported the median value. Fig. 5(b) and (c) shows the effects of different shrinkage scales on the watermark correlation value and PSNR value, respectively. From Fig. 5(b) and (c), it is clear that the best choice of shrinkage scale is 0.5.

In order to demonstrate how robust our watermark is, we attacked the embedded watermark by means of randomization of vertex coordinates. To simulate such attacks, randomization of vertex coordinates was controlled by means of the noise strength, which is defined as the ratio of the largest displacement to the longest edge of the object's bounding box. In this analysis, we picked five models with the largest resolution level from the set of analysis models and fixed the shrinkage scale at 0.5. With various bin sizes, our watermark was embedded into each model and then attacked using different noise strengths in robustness tests. In the same way, we ran each test five times using different keys and reported the median value. Fig. 5(d) shows the results of robustness tests using different bin sizes for the HIV-lv5 model. From these plots, it can be seen that a larger bin size can provide a hidden watermark with higher robustness. However, the drawback is that the false-alarm rate is increased as well.

IV. Experimental Results

A series of experiments were conducted to test the performance of the proposed fragile watermarking method. We shall start with parameter selection and then report quantitatively some experimental results. In addition, we shall present a set of visualization results that can demonstrate the power of the proposed method in distinguishing malicious attacks from incidental modifications.

A. Selecting Appropriate Parameters

We have reported in Section III that several parameters were needed during watermark embedding and detection. These parameters included a binary watermark pattern, a set of LUTs, a basis for parameterization, and the degree of quantization. All of the parameters used in our experiments were set as follows. A binary watermark pattern with a size of 512×512 (as indicated in Fig. 6) was used in our experiments. That means, $WM\_X\_SIZE = WM\_Y\_SIZE = 512$. In addition, a set of LUTs were generated and protected by one authentication key. The size of each table was 256. Therefore, $LUT\_SIZE = 256$. As to the basis for parameterization, since the 3-D vertex space is periodically aggregated into binary state spaces, its selection is not crucial to the proposed method. Therefore, we fixed the basis as $m(0,0,0)$ and $n(1,0,0)$ in the experiments. As for appropriate quantization steps, we assigned the ordinary numeric value, $b = 0.2$, in all the experiments such that the performance of our method is close to optimal (i.e., $C \cong 1$). The selection of $b = 0.2$ was based on the experiments gained from conducting quite a number of experiments. However, since the selection is an ill-posed problem, it is hard to systematically determine a right value that can fit in all cases.

One thing to be noted is that the basis (m, n) and the quantization step b together can possibly be hard-coded into the algorithm so that detecting a watermark for the purpose of authentication can be realized as oblivious detection. However, the drawback is that the robustness (i.e., the domain of acceptable alternation) certainly varies with applications. These are the restriction that are associated with an LUT/secret key approach in general.

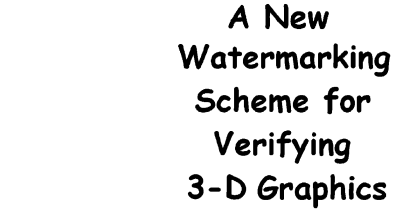

Fig. 6. Binary watermark pattern used in our experiments.

TABLE II
List of Five Triangulated Meshes Used in Our Experiments and Their Watermark Correlation Values Detected Using the Proposed Method

Model	Number of vertices/faces	Correlation value
dolphins	855/1692	1
spock	16386/32768	1
mannequin	711/1418	1
holes3	5884/11776	1
HIV	9988/20000	1

B. Experimental Results of Authentication

The data set used in our experiments was a set of triangulated and closed meshes, listed in Table II. Each of them was watermarked using our fragile watermarking method presented in Section III. The last column in Table II shows the watermark correlation values for the five different models. The five test models were watermarked and tested to evaluate the robustness against reduction of floating-point precision. The results of this experiment are shown in Fig. 7, where the precision of a floating-point number is specified by a nonnegative decimal integer preceded by a period (.) and succeeded by a character f. It is clearly shown in Fig. 7 that the proposed method is very robust against vertex quantization down to three decimal digits. Note that for authentication applications one has to rely on visual inspection since the correlation coefficient does not signal. For instance, for meshes with a large number of vertices, only modifying a small region does not affect the correlation value substantially. In addition, finding a threshold that is suitable for all kinds of meshes and resolutions is very difficult. In what follows, therefore, we shall show how to visualize the authentication results.

C. Visualization of Authentication Results

Visualization is a good way to "see" whether the proposed watermarking method is valid or not. Fig. 8 shows that the original and the watermarked Spock models were rendered as either wireframe or flat-shaded models, respectively. It can be seen that the watermarked model maintained high correlation with the original model, whether in a wireframe format or in a flat-shaded format.

The results of experiments on detecting malicious attacks from some incidental modifications are shown in Figs. 9 and

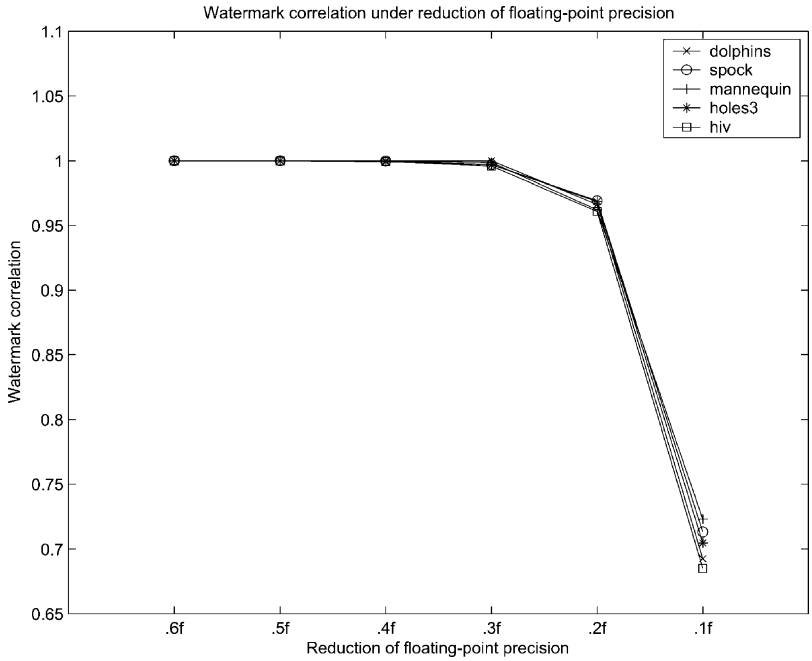

Fig. 7. Five test models were watermarked and tested to evaluate the robustness against reduction of floating-point precision.

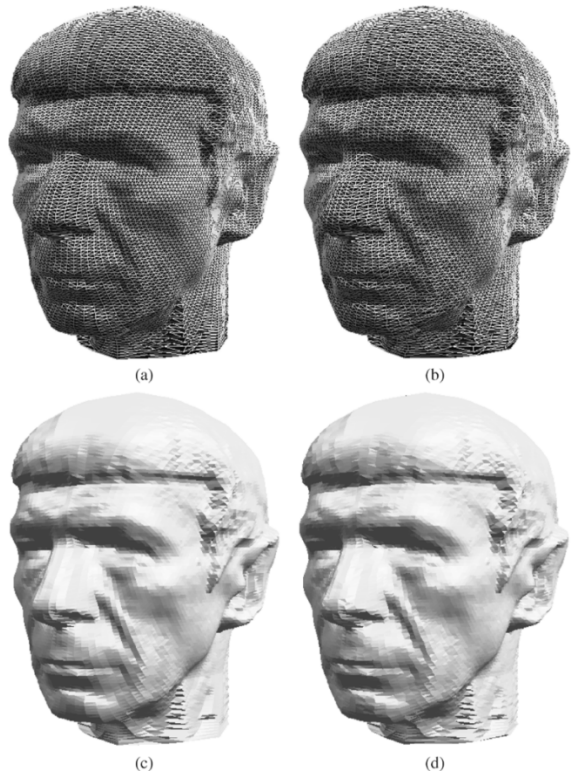

Fig. 8. Visualization of the transparency test: (a) original Spock model rendered in a wireframe format; (b) watermarked Spock model rendered in a wireframe format; (c) original Spock model rendered in a flat-shaded form; and (d) watermarked Spock model rendered in a flat-shaded form.

10. Fig. 9(a) shows that the watermarked Spock model was tampered with by stretching out Spock's nose. In addition, the quantization down to two decimal digits was applied to the vertex coordinates of the watermarked Spock model that has been tampered with. Fig. 9(b) shows some detected potentially modified

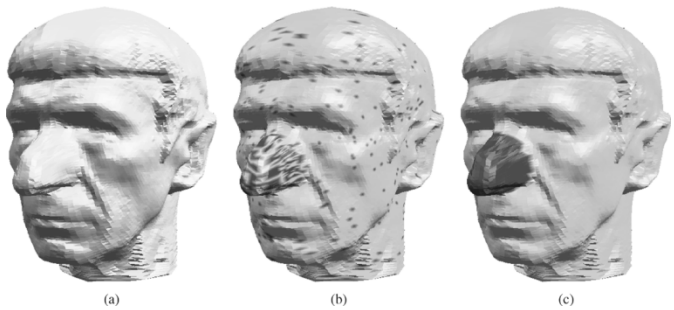

Fig. 9. Region-based tampering detection: (a) watermarked Spock model tampered with by stretching out its nose, which was followed by applying the quantization (down to two decimal digits) to the vertex coordinates.; (b) detected potentially modified regions (before morphological operators were applied); and (c) detected modified regions after the morphological operators were applied.

regions before the closing operator was applied. Note that approximately 50% of vertices on Spock's nose were identified as invalid vertices, as shown in Fig. 9(b). Therefore, in order to amplify the effect of the authentication results, the morphological operators described in [29] were adopted so that the parts being tampered with in a model could be detected and highlighted. Fig. 9(c) shows the authentication results of Fig. 9(b) after some morphological operations were applied. Fig. 10 shows another example of malicious tampering involving vertex quantization, which could possibly occur in the real world. In this case, it is not obvious that the two dolphins were tampered with. Nevertheless, the proposed method still succeeded in malicious tampering detection. As shown in Fig. 10(d), among the two dolphins that were tampered with, one was translated, and the other one stretched out. Both attacks were detected and highlighted.

V. Conclusion

A new fragile watermarking scheme which can be applied to authenticate 3-D polygonal meshes has been presented in this

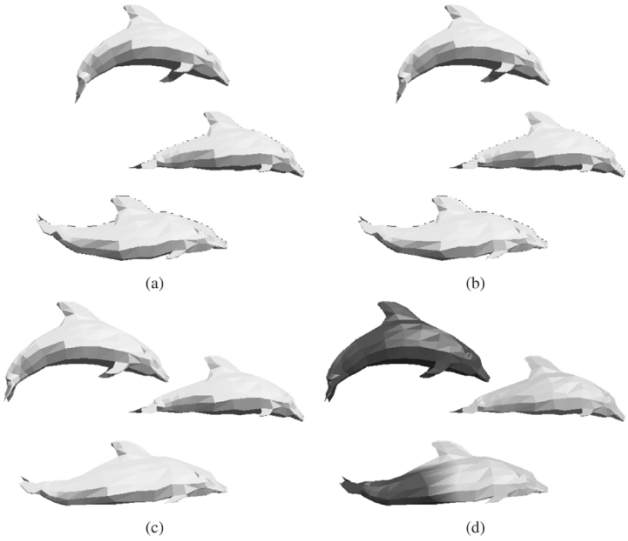

Fig. 10. Detection of malicious attack involving the incidental modification (such as quantization of vertex coordinates): (a) original dolphins model; (b) watermarked dolphins model; (c) slightly modified dolphins model; and (d) two out of the three dolphins have been tampered with. The maliciously modified dolphins were effectively detected.

paper. Watermarks are embedded using a local mesh parameterization technique and can be blindly extracted for authentication applications. The proposed scheme has three remarkable features: 1) the domain of allowable alternation for a vertex is explicitly defined by two well-designed hash functions; 2) region-based tampering detection is achieved by a vertex-order-independent embedding process; and 3) fragile watermarking is achieved for localization of malicious modifications and tolerance of certain incidental manipulations (such as quantization of vertex coordinates and vertex reordering). To the best of our knowledge, this is the first 3-D mesh authentication scheme that can detect malicious attacks involving certain incidental modifications.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their comments and suggestions which have improved the readability and technical content of this paper. Polygonal meshes used in this paper were provided courtesy of the University of Washington and Cyberware. For the use of the HIV protease surface model, the authors would like to thank A. Olson of The Scripps Research Institute.

REFERENCES

[1] N. Aspert, E. Drelie, Y. Maret, and T. Ebrahimi, "Steganography for three-dimensional polygonal meshes," *Proc. SPIE*, vol. 4970, 2002.

[2] O. Benedens, "Affine invariant watermarks for 3-D polygonal and NURBS based models," in *Proc. Information Security Workshop*, Wollongon, Australia, 2000, pp. 20–21.

[3] ——, "Robust watermarking and affine registration of 3-D meshes," in *Proc. Information Hiding*, Noordwijkerhout, The Netherlands, 2002, pp. 177–195.

[4] P. J. Burt and E. H. Adelson, "Laplacian pyramid as a compact image code," *IEEE Trans. Commun.*, vol. COM-31, pp. 532–540, 1983.

[5] H. Biermann, I. Martin, F. Bernardini, and D. Zorin, "Cut-and-paste editing of multiresolution surfaces," in *Proc. SIGGRAPH*, San Antonio, TX, 2002, pp. 312–321.

[6] F. Cayre and B. Macq, "Data hiding on 3-D triangle meshes," *IEEE. Trans. Signal Process.*, vol. 51, no. 4, pp. 939–949, Apr. 2003.

[7] F. Cayre, P. Rondao-Alface, F. Schmitt, B. Macq, and H. Maître, "Application of spectral decomposition to compression and watermarking of 3-D triangle mesh geometry," *Signal Process.: Image Commun.*, vol. 18, pp. 309–319, 2003.

[8] M. Desbrun, M. Meyer, P. Schröder, and A. Barr, "Implicit fairing of irregular meshes using diffusion and curvature flow," in *Proc. SIGGRAPH*, Los Angeles, CA, 1999, pp. 317–324.

[9] M. S. Floater, "Parameterization and smooth approximation of surface triangulations," *Comput. Aided Geom. Design*, vol. 14, pp. 231–250, 1997.

[10] C. Fornaro and A. Sanna, "Public key watermarking for authentication of CSG models," *Comput.-Aided Design*, vol. 32, pp. 727–735, 2000.

[11] M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," in *Proc. SIGGRAPH*, Los Angeles, CA, 1997, pp. 209–216.

[12] I. Guskov, W. Sweldens, and P. Schröder, "Multiresolution signal processing for meshes," in *Proc. SIGGRAPH*, Los Angeles, CA, 1999, pp. 325–334.

[13] H. Hoppe, "Progressive meshes," in *Proc. SIGGRAPH*, New Orleans, LA, 1996, pp. 99–108.

[14] A. Johnson and M. Hebert, "Using spin-images for efficient multiple model recognition in cluttered 3-D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.

[15] ——, "Control of polygonal mesh resolution for 3-D computer vision," *Graph. Models and Image Process.*, vol. 60, pp. 261–285, 1998.

[16] B. Koh and T. Chen, "Progressive browsing of 3-D models," in *Proc. IEEE Signal Processing Soc. Workshop on Multimedia Signal Processing*, Copenhagen, Denmark, 1999, pp. 71–76.

[17] M. S. Kankanhalli, E.-C. Chang, X. Guan, Z. Huang, and Y. Wu, "Authentication of volume data using wavelet-based foveation," in *Proc. 6th Eurographics Workshop on Multimedia*, Manchester, U.K., 2001.

[18] S. Kanai, H. Date, and T. Kishinami, "Digital watermarking for 3-D polygons using multiresolution wavelet decomposition," in *Proc. 6th IFIP WG 5.2 GEO-6*, Tokyo, Japan, 1998, pp. 296–307.

[19] Z. Karni and C. Gotsman, "Spectral compression of mesh geometry," in *Proc. SIGGRAPH*, New Orleans, LA, 2000, pp. 279–286.

[20] A. W. F. Lee, W. Sweldens, P. Schröder, L. Cowsar, and D. Dobkin, "MAPS: Multiresolution Adaptive Parameterization of Surfaces," in *Proc. SIGGRAPH*, Orlando, FL, 1998, pp. 95–104.

[21] X. Mao, M. Shiba, and A. Imamiya, "Watermarking 3-D geometric models through triangle subdivision," *Proc. SPIE*, vol. 4314, pp. 253–260, 2001.

[22] R. Ohbuchi, H. Masuda, and M. Aono, "Watermarking three-dimensional polygonal models through geometric and topological modifications," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 551–560, 1998.

[23] ——, "Data embedding for geometrical and nongeometrical targets in three-dimensional polygonal models," *Comput. Commun.*, vol. 21, pp. 1344–1354, 1998.

[24] ——, "A shape-preserving data embedding algorithm for NURBS curves and surfaces," in *Proc. Computer Graphics Int.*, Canmore, AB, Canada, 1999, pp. 177–180.

[25] R. Ohbuchi, A. Mukaiyama, and S. Takahashi, "A frequency-domain approach to watermarking 3-D shapes," in *Proc. EUROGRAPHICS*, vol. 21, Saarbrucken, Germany, 2002, pp. 373–382.

[26] R. Ohbuchi, S. Takahashi, and T. Miyazawa, "Watermarking 3-D polygonal meshes in the mesh spectral domain," in *Proc. Graphics Interface*, ON, Canada, 2001, pp. 9–17.

[27] E. Praun, H. Hoppe, and A. Finkelstein, "Robust mesh watermarking," in *Proc. SIGGRAPH*, Los Angeles, CA, 1999, pp. 154–166.

[28] E. Praun, W. Sweldens, and P. Schröder, "Consistent mesh parameterizations," in *Proc. SIGGRAPH*, Los Angeles, CA, 2001, pp. 179–184.

[29] C. Rössl, L. Kobbelt, and H. P. Seidel, "Extraction of feature lines on triangulated surfaces using morphological operators," in *Symp. Smart Graphics*, Stanford, CA, 2000.

[30] G. Taubin, "A signal processing approach to fair surface design," in *Proc. SIGGRAPH*, Los Angeles, CA, 1995, pp. 351–358.

[31] G. Turk, "Re-tiling polygonal surfaces," in *Proc. SIGGRAPH*, Chicago, IL, 1992, pp. 55–64.

[32] ——, "Generating random points in triangles," in *Graphics Gems*, A. Glassner, Ed. New York: Academic, 1999.

[33] K. Yin, Z. Pan, S. Jiaoying, and D. Zhang, "Robust mesh watermarking based on multiresolution processing," *Comput. and Graph.*, vol. 25, pp. 409–420, 2001.

[34] B. L. Yeo and M. M. Yeung, "Watermarking 3-D objects for verification," *IEEE Comput. Graph. Applic.*, vol. 19, pp. 36–45, 1999.

[35] M. M. Yeung and B. L. Yeo, "An invisible watermarking technique for image verification," in *Proc. Int. Conf. Image Processing*, vol. 2, Piscataway, NJ, 1997, pp. 680–683.
[36] M. G. Wagner, "Robust watermarking of polygonal meshes," in *Proc. Geometric Modeling and Processing*, Hong Kong, 2000, pp. 201–208.
[37] W. Sweldens and P. Schröder, "Course 50: digital geometry processing," in *SIGGRAPH'2001 Course Note*, 2001.

Hsueh-Yi Sean Lin was born in Taipei, Taiwan, R.O.C., in 1975. He received the B.S. degree from the Chung-Hua University, Hsinchu, Taiwan, in 1997, and the M.S. degree from the Yuan-Ze University, Chung-Li, Taiwan, in 1999, all in computer science. He is currently pursuing the Ph.D. degree in the Department of Computer and Information Science, National Chiao-Tung University, Hsinchu.

His current research interests include 3-D mesh processing, retrieval, and authentication.

Hong-Yuan Mark Liao (SM'01) received the B.S. degree in physics from National Tsing-Hua University, Hsinchu, Taiwan, R.O.C., in 1981, and the M.S. and Ph.D. degrees in electrical engineering from Northwestern University, Evanston, IL, in 1985 and 1990, respectively.

He was a Research Associate with the Computer Vision and Image Processing Laboratory, Northwestern University, during 1990–1991. In July 1991, he joined the Institute of Information Science, Academia Sinica, Taipei, Taiwan, as an Assistant Research Fellow. He was promoted to Associate Research Fellow and then Research Fellow in 1995 and 1998, respectively. From August 1997 to July 2000, he served as the Deputy Director of the institute. From February 2001 to January 2004, he was the Acting Director of the Institute of Applied Science and Engineering Research. He is jointly appointed as a Professor of the Computer Science and Information Engineering Department of National Chiao-Tung University. His current research interests include multimedia signal processing, wavelet-based image analysis, content-based multimedia retrieval, and multimedia protection.

Dr. Liao is the Managing Editor of the *Journal of Information Science and Engineering*. He is on the Editorial Board of the *International Journal of Visual Communication and Image Representation*, *Acta Automatica Sinica*, and the EURASIP *Journal on Applied Signal Processing*. He was an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA during 1998–2001. He received the Young Investigators' award from Academia Sinica in 1998; the Excellent Paper Award from the Image Processing and Pattern Recognition society of Taiwan in 1998 and 2000, the Distinguished Research Award from the National Science Council of Taiwan in 2003, and the National Invention Award in 2004. He served as the Program Chair of the International Symposium on Multimedia Information Processing (ISMIP'97), the Program co-chair of the Second IEEE Pacific-Rim conference on Multimedia (2001), and the Conference co-chair of the Fifth IEEE International Conference on Multimedia and Exposition (ICME).

Chun-Shien Lu (M'99) received the Ph.D. degree in electrical engineering from National Cheng-Kung University, Tainan, Taiwan, R.O.C., in 1998.

From October 1998 to July 2002, he joined Institute of Information Science, Academia Sinica, Taiwan, as a postdoctoral fellow for his military service. Since August 2002, he has been an Assistant Research Fellow at the same institute. His current research interests mainly focus on multimedia technologies and applications.

Dr. Lu organized a special session on Multimedia Security in the 2nd and 3rd IEEE Pacific-Rim Conference on Multimedia, respectively (2001–2002). He co-organized two special sessions (in the area of media identification and DRM) in the 5th IEEE International Conference on Multimedia and Expo (ICME), 2004. He is a guest co-editor of EURASIP *Journal on Applied Signal Processing* special issue on Visual Sensor Network in 2005. He has two U.S. patents, two R.O.C. patents, and one Canadian patent in digital watermarking. He has received the paper awards many times from the Image Processing and Pattern Recognition society of Taiwan for his work on data hiding.He was a co-recipient of a National Invention and Creation Award in 2004. He is a member of the ACM.

Ja-Chen Lin was born in Taiwan, R.O.C., in 1955. He received the B.S. degree in computer science in 1977 and the M.S. degree in applied mathematics in 1979, both from National Chiao-Tung University (NCTU), Hsinchu, Taiwan, and the Ph.D. degree in mathematics from Purdue University, West Lafayette, IN, in 1988.

From 1981 to 1982, he was an Instructor at NCTU. From 1984 to 1988, he was a Graduate Instructor at Purdue University. He joined the Department of Computer and Information Science, NCTU, in August 1988, and is currently a Professor there. His recent research interests include pattern recognition and image processing.

Dr. Lin is a member of the Phi Tau Phi Scholastic Honor Society.