A Comprehensive Survey on Three-Dimensional Mesh Watermarking

Kai Wang*, Guillaume Lavoué, Florence Denis, and Atilla Baskurt

Abstract

Three-dimensional meshes have been used more and more in industrial, medical and entertainment applications during the last decade. Many researchers, from both the academic and the industrial sectors, have become aware of their intellectual property protection and authentication problems arising with their increasing use. This paper gives a comprehensive survey on 3D mesh watermarking, which is considered an effective solution to the above two emerging problems. Our survey covers an introduction to the relevant state of the art, an attack-centric investigation, and a list of existing problems and potential solutions. First, the particular difficulties encountered while applying watermarking on 3D meshes are discussed, followed by a presentation and an analysis of the existing algorithms, distinguishing them between fragile techniques and robust techniques. Since the attacks play an important role in the design of 3D mesh watermarking algorithms, we also provide an attack-centric viewpoint of this state of the art. Finally, some future working directions are pointed out especially on the ways of devising robust and blind algorithms and on some new probably promising watermarking feature spaces.

Index Terms

3D mesh, digital watermarking, copyright protection, authentication, attack, robustness.

I. Introduction

Digital watermarking [1]–[3] has been considered a potential efficient solution for copyright protection of various multimedia contents. This technique carefully hides some secret information in the functional part of the cover content. Compared with traditional cryptography, the digital watermarking technique is able to protect digital works (assets) after the transmission phase and the legal access. There exist different classifications of watermarking algorithms. We distinguish between *non-blind* and *blind* watermarking

K. Wang* (corresponding author), G. Lavoué, and A. Baskurt are with Laboratory LIRIS, UMR 5205 CNRS, INSA-Lyon, Villeurbanne, F-69621 France. (e-mails: {kwang, glavoue, abaskurt}@liris.cnrs.fr, phone: 33-472436097, fax: 33-472437117)

F. Denis is with Laboratory LIRIS, UMR 5205 CNRS, Université Lyon 1, Villeurbanne, F-69622 France. (e-mail: fdenis@liris.cnrs.fr)

schemes depending on whether or not the original digital work is needed at extraction. Usually, one hopes to construct a *robust* watermark, which is able to go through common malicious attacks, for copyright protection purposes. However, sometimes the watermark is intentionally designed to be *fragile*, even to very slight modifications, in order to be used in authentication applications. Finally, researchers customarily classify watermarking algorithms into two categories, *spatial-domain-based* or *transform-domain-based*, according to the insertion space.

Nowadays, 3D meshes are widely used in virtual reality, medical imaging, video games and computer aided design. A mesh is a collection of polygonal facets targeting to constitute an appropriate approximation of a real 3D object. It owns three different combinatorial elements: *vertices*, *edges*, and *facets*. From another viewpoint, a mesh can also be completely described by two kinds of information: the *geometry* information describes the 3D positions (coordinates) of all its vertices, while the *connectivity* information provides the adjacency relations between the different elements. Mathematically, a 3D polygonal mesh containing N vertices and M edges can be modeled as a signal $M = \{G, C\}$, where

$$G = \{v_i\}_{i=1,2,...,N}, v_i = (x_i, y_i, z_i) \tag{1}$$

$$C = \{(v_{k_1}, v_{k_2})\}, 1 \leq k_1 \leq N, 1 \leq k_2 \leq N, k_1 \neq k_2 \tag{2}$$

Each vertex element v_i in G is numbered by an index i and is described by its three-dimensional coordinates (x_i, y_i, z_i); C has M elements and each element stands for an edge connecting two different vertices indexed by k_1 and k_2, respectively. In practice, instead of a list of edges, users usually prefer a list of all the mesh facets with their respective component vertices in a certain cyclic order. Although this list contains redundant information, it can facilitate the geometrical and topological operations on a given mesh. Fig. 1 shows an example of a 3D mesh. As illustrated by the close-up, the *degree* of a facet is the number of its component edges, and the *valence* of a vertex is defined as the number of its incident edges. Their formal mathematical definitions are given at the end of this paragraph. Although there are many other 3D representations, such as cloud of points, parametric surface, implicit surface and voxels, 3D mesh has become the *de facto* standard for numerical representation of 3D objects thanks to its algebraic simplicity and usability. Furthermore, it is quite easy to convert other representations to 3D mesh, which is considered a low-level, but effective model.

Definition 1 (Degree of a facet): A facet is a minimum cycle of edges on the mesh surface that does not contain any other edge cycles. Formally, a facet f_i can be defined as a sequence of vertices $\{v_{i_1}, v_{i_2}, ..., v_{i_J}\}$, where $v_{i_j} \in G, j = 1, 2..., J$. The degree of the facet f_i is defined as the number of its component edges, and it is easy to deduce that this number is simply J (also the number of its component

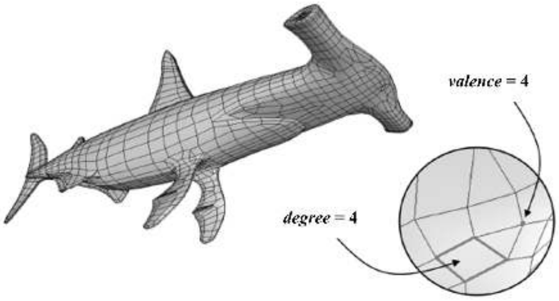

Fig. 1. Example of a 3D mesh and a close-up illustrating the *valence* of a *vertex* and the *degree* of a *facet*.

vertices) in the above expressions. Furthermore, we often request that all the vertices forming a facet should be on exactly the same plane.

Definition 2 (Valence of a vertex and its 1-ring neighbors): The valence of the vertex v_i is defined as:

$$valence(v_i) = \mathcal{N}(\{v_j | v_j \in G,\ v_j \neq v_i, and\ (v_i, v_j) \in C\}) \tag{3}$$

where $\mathcal{N}(.)$ is a function that returns the cardinality of a set, and all the elements of the set $\{v_j | v_j \in G,\ v_j \neq v_i, and\ (v_i, v_j) \in C\}$ form the 1-ring neighbors of the vertex v_i.

With the increasing capability of capturing, processing and visualizing 3D data, the intellectual property protection of 3D meshes has attracted more and more attention. Naturally, as a promising technique, watermarking appears to be a good candidate for solving this increasing problem. Fragile watermarks can also be used to authenticate the origin and integrity of the received 3D mesh data at the user end. This paper, as the extended version of [4], reviews the nearly 10-year history of the research on 3D mesh watermarking since the publication of the first relevant algorithms in 1997 [5], and it provides some suggestions on the future working directions in this developing discipline. The remainder of this paper is organized as follows. Section II discusses the special difficulties encountered when watermarking 3D meshes and provides an overview of the most important techniques proposed in the literature. Attacks on watermarked meshes play an important role in the design of suitable watermarking algorithms. They are much more intractable than their counterparts on images. Therefore, section III is dedicated to analyzing various possible attacks and discussing the corresponding solutions in order to resist them. Some open questions and possible research directions are detailed in the last section.

II. 3D Mesh Watermarking Techniques

A. Difficulties and Classification

There still exist few watermarking methods for 3D meshes, in contrast with the relative maturity of the theory and practices of image, audio and video watermarking. This situation is mainly caused by the difficulties encountered while handling the arbitrary topology and irregular sampling of 3D meshes, as well as the complexity of the possible attacks on watermarked meshes.

We can consider an image as a matrix, and each pixel as an element of this matrix. This means that all of these pixels have an intrinsic order in the image, for example the order established by row or column scanning. This order is usually used to synchronize watermark bits. On the contrary, there is no simple robust intrinsic ordering for mesh elements, which often constitute the watermark bit carriers (primitives). Some intuitive orders, such as the order of the vertices and facets in the mesh file, and the order of vertices obtained by ranking their projections on an axis of the objective Cartesian coordinate system, are easy to be altered. In addition, because of their irregular sampling, we still lack an effective spectral analysis tool for 3D meshes. This situation, as you can see in the following, makes it difficult to apply existing successful spectral watermarking schemes, such as the one proposed in [6], on 3D meshes.

In addition to the point raised above, robust watermarks also have to face various intractable attacks. The reordering of vertices and facets do not have any impact on the shape of the mesh, while it can seriously desynchronize the watermarks that rely on this straightforward ordering. The similarity transformations, including translation, rotation and uniform scaling, are supposed to be common operations through which a robust watermark should survive. Even worse, the original watermark primitives can disappear after a mesh simplification or remeshing. Such tools are available in much software, and they can completely destroy the connectivity information of the watermarked mesh while well conserving its shape. Usually, we distinguish between geometric attacks, which only modify the positions of the vertices, and connectivity attacks, which also change the connectivity aspect. Section III provides a detailed investigation on these attacks and discusses the existing solutions to make the watermarks robust against them.

Watermarking 3D meshes in computer aided design applications has other difficulties caused by design constraints. For example, the symmetry of the object has to be conserved and the geometric modifications have to be within a tolerance for future assembly. Under this situation, the watermarked mesh will no longer be evaluated only by the human visual system, which is quite subjective, but also by some strict objective metrics.

In the following, we introduce the existing 3D mesh watermarking algorithms by distinguishing them between fragile techniques and robust techniques. In each class, it seems convenient to subdivide the members into two subclasses, depending on whether the watermark is embedded in the spatial domain (by modifying the geometry or the connectivity) or in a transform domain (by modifying the coefficients obtained after a certain transformation).

B. Fragile Techniques

A fragile technique for authentication application often has to possess two features: it should be vulnerable to even very slight modifications of the watermarked asset; and it should be capable of locating, or even identifying the endured attacks. In this subsection, we will also mention some high-capacity but fragile techniques used for annotation or covert communication applications.

1) Fragile Techniques in Spatial Domain: As stated before, the spatial description of a 3D mesh includes a geometry aspect and a connectivity aspect. We first present the techniques modifying the geometry.

- *Fragile Techniques in Spatial Domain Modifying the Geometry*

The algorithms that directly modify the individual vertices' positions are often fragile techniques. Yeo and Yeung [7] proposed such a fragile algorithm that can be used for mesh authentication. The basic idea is to search for a new position for each vertex where two predefined hash functions have an identical value, in order to make all vertices valid for authentication. At the extraction phase, one simply examines the validity of each vertex and locates the possible attacks on the invalid vertices. The watermark embedding algorithm depends on a pre-established vertex order to prevent the causality problem. Formally, *causality problem* means that the insertion of the posterior watermark bits impacts the synchronization of the anterior inserted watermark bits, or directly changes the feature values of the watermark primitives where the anterior bits are inserted; hence, the extracted bits can be different from the original ones, even in the absence of attacks. Here in the algorithm of Yeo and Yeung [7], the first hash function is dependent only on the position of the current vertex to be watermarked, but the second one is also dependent on the positions of the 1-ring neighbors of this current vertex. When considering the 1-ring neighbors for hash function calculation, the authors only take into account the already watermarked ones, which are in front of the current vertex in the pre-established order. Without this order, the causality problem occurs, which in this case means that the watermarking of one vertex can impact the validities of its neighbors that have already been watermarked. Hence, the proposed scheme is fragile to vertex rearrangement.

Lin *et al.* [8] considered vertex rearrangement as an operation that even a fragile watermark should be able to resist because it is harmless to the mesh shape. Thus, they solved the causality problem by setting both hash functions dependent only on the coordinates of the current vertex. They also proposed a more controllable modification scheme with a better attack localization capability. Chou and Tseng [9] solved the causality problem by introducing the adjusting vertex method. In their watermarking algorithm,

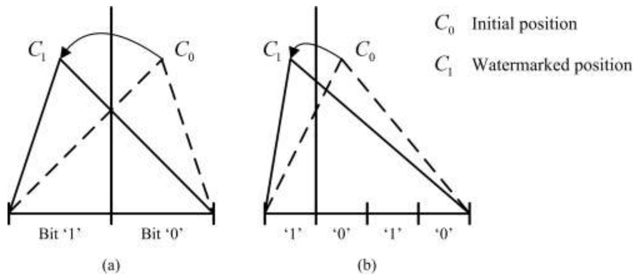

Fig. 2. Watermarking primitive in the algorithm of Cayre and Macq [10], the projection of a vertex is moved into the nearest correct interval: (a) the opposite edge is divided into two intervals; (b) the opposite edge is divided into four intervals. The inserted bits are both '1'.

one of the two hash functions is dependent on the barycenter of the vertex 1-ring neighbors. However, nearly every watermarked vertex owns an adjusting vertex selected from its neighbors. The position of the adjusting vertex is tuned in order to keep the barycenter of the neighbors of the watermarked vertex unchanged after watermarking. Another feature is that the distortion up-bound for each watermarked vertex is accurately controlled so that severe distortions, which are possible in [7], [8], are avoided.

High-capacity watermarks are often fragile, and that is the reason for why we present them here, in fragile techniques section. In practice, individual vertex coordinates are also used to construct high-capacity steganographic approaches. Cayre and Macq [10] proposed a high-capacity blind data-hiding algorithm for triangular meshes. By choosing the projection of a vertex on its opposite edge in a triangle as the primitive (see Fig. 2), the theoretical capacity can attain 1 bit per vertex. The synchronizing mechanism relies on the choice of the first triangle according to a certain geometrical criterion (*e.g.* one of the triangles intersecting with the most significant principal axis of the mesh) and a further geometric spreading scheme that is piloted by a secret key. Bors [11] also reported a blind algorithm. The primitive is the relative position of a vertex to its 1-ring neighbors. A two-state space division is established (*e.g.* inside or outside of an ellipsoid), and the vertex is assumed to be moved into the correct subspace according to the next watermark bit. A higher capacity, which is about 3 bits per vertex, is achieved in [12] by applying a multi-level embedding procedure. This procedure consists of modifying successively the parallel, vertical, and rotary positions of a vertex related to its opposite edge in a triangular facet. By quantizing the distance of a facet to the mesh center, Wu and Chueng [13] gave another fragile but high-capacity scheme whose capacity can attain 1 bit per facet.

It is worth pointing out that for the fragile techniques used for authentication (integrity verification), the researchers focus more on how to solve the causality problem, than on how to obtain the robustness (or rather invariance) against similarity transformations. These transformations have theoretically no impact on the mesh shape, but could cause slight perturbations of the mesh geometric description due to calculation and storage precision limits. In many authentication applications, these transformations are not considered as malicious attacks, but as routine operations. Indeed, most existing high-capacity methods are invariant to these transformations, but they are not able to precisely locate the attacks. The reason for this is that it is either very difficult to have all the mesh combinatorial elements watermarked [10], or it is due to the desynchronization problem [11]. The latter means that the error of a certain bit can make all the posterior extraction incredible, thus failing to authenticate the rest parts of the mesh.

- *Fragile Techniques in Spatial Domain Modifying the Connectivity*

Presently, there are only a few 3D mesh watermarking techniques based on connectivity modification (neither fragile nor robust). Ohbuchi *et al.* [5] presented two such algorithms. In the first one, the local triangulation density is changed to insert a visible watermark. The second algorithm first cuts one band of triangular facets off the mesh and then glues it to the mesh with just one edge. This facet band can be a meaningful pattern or simply determined by a secret key. Both methods are visible and fragile. But the embedded watermarks do not spread all over the mesh, and this fact stops them from being useful fragile watermarks for integrity authentication due to the lack of attack localization capability.

2) Fragile Techniques in Transform Domain: Usually, researchers insert watermarks in a kind of spectral domain of the asset to improve the robustness or the imperceptibility, according to the spread spectrum communication principle. However, some other transformations, such as multiresolution analysis, are much more flexible.

Practically, 3D mesh multiresolution analysis [14] is a useful tool to reach an acceptable trade-off between the mesh complexity and the capacity of the available resources. Such an analysis finally produces a very coarse mesh that represents the basic shape (low frequencies) and a set of details information at different resolution levels (median and high frequencies). These methods also permit realizing a synthesis process during which multiple representations with different complexities (*i.e.* resolutions) can be created.

As mentioned above, the most interesting point of the multiresolution analysis for watermarking is its flexibility: there are different available locations authorized to meet different application demands. For example, insertion in the coarsest-level representation ensures a good robustness, while embedding in the details parts provides an excellent capacity. Under the same additive insertion intensity, insertion in the mesh low resolution component can be both more robust and more imperceptible because such an insertion makes the object expand or contract a little, while keeping its basic shape. The insertion in high resolution levels may permit constructing some effective fragile watermarks with a precise localization ability of the attacks.

Wavelets are a common tool for such a multiresolution analysis. The mathematical formulation of the

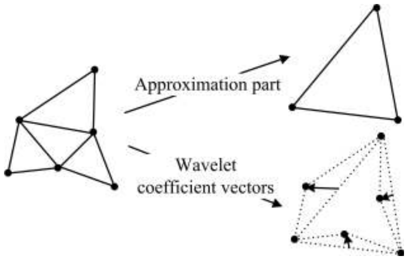

Fig. 3. Illustration of one iteration of the lazy wavelet decomposition mechanism of 3D semi-regular triangular meshes.

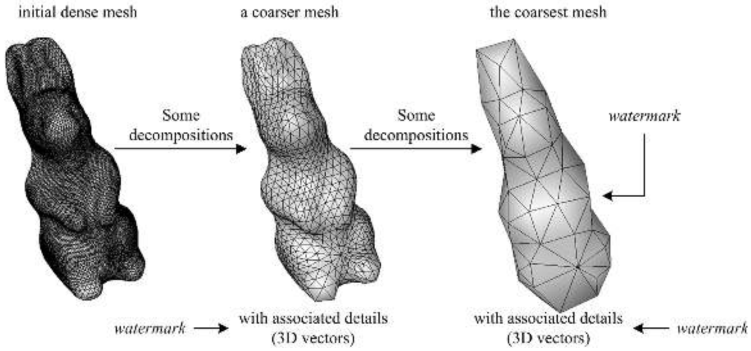

Fig. 4. 3D mesh watermarking techniques based on wavelet transform.

wavelet analysis and synthesis of 3D meshes was introduced by Lounsbery *et al.* [15]. Fig. 3 illustrates one iteration of the lazy wavelet decomposition mechanism. A group of four triangles is merged into one, and three of the six initial vertices are conserved in the lower resolution. The wavelet coefficients are calculated as the prediction errors for all the deleted vertices, and they are 3D vectors associated with each edge of the coarser mesh. One straightforward prediction is the midpoint of the two conserved vertices having been incident to the deleted vertex. Note that this kind of wavelet analysis is applicable only on semi-regular triangular meshes. Fig. 4 shows the wavelet decomposition of a dense rabbit mesh; the watermark can be inserted either in the coarsest-level mesh (robust watermark), or in the wavelet coefficients at different levels (high-capacity or fragile watermarks).

Cho *et al.* [16] proposed a fragile algorithm in the wavelet domain to authenticate semi-regular meshes. They first apply several wavelet decompositions on the original triangular mesh and then consider the facets in the obtained coarser mesh as authentication primitives. The basic idea is to slightly modify each facet so that the values of two predefined functions are the same, in order to make all these facets valid for authentication. Both function inputs are invariant to similarity transformations. However, it seems that two problems exist: first, the causality problem occurs because the modification of the current to-be-watermarked facet can influence the validities of its already watermarked neighboring facets, and this problem is not mentioned by the authors; secondly, the watermark is inserted in a relatively coarse mesh obtained after several wavelet decompositions, which seems disadvantageous to provide precise attack localization capability. Recently, Wang *et al.* [17] have described a fragile watermarking technique for authenticating semi-regular meshes that is both robust to vertex reordering and similarity transformations, and capable of precisely locating the endured attacks. In their method, after one wavelet decomposition, the norm and the orientation of each obtained wavelet coefficient vector are independently modified so as to make them both imply a same watermark symbol, serving for authentication. Based on wavelet analysis, a high-capacity scheme is reported in [18], which relies on modification of the norms' permutation of the wavelet coefficient vectors at a same resolution level.

C. Robust Techniques

A robust technique should at least be able to resist the attacks that cause distortions smaller than a certain threshold beyond which the watermarked mesh is greatly degraded. However, we always hope to construct robust techniques as strong as possible while keeping the watermark imperceptible.

1) Robust Techniques in Spatial Domain: Between the geometry and the connectivity parts of a 3D mesh, nearly all the existing spatial robust algorithms take the former as primitive, which shows superiority in both robustness and imperceptibility compared to the latter. The fragility to connectivity attacks of the algorithms modifying connectivity information prevents them from being robust watermarks. It is important to note that this section focuses more on watermarking primitives than on robustness, which will be explored in detail in the next section. Also note that some techniques presented in this subsection are not strictly robust, but they are neither fragile. Actually, these techniques have witnessed the development of 3D mesh watermarking techniques from simple data hiding schemes to robust and blind algorithms.

As reported in the subsection concerning fragile techniques in spatial domain, inserting 1 bit in each vertex makes the algorithms very vulnerable. Therefore, some algorithms choose the positions of groups of vertices as watermarking primitives in order to try to strengthen the robustness. Yu *et al.* [19] gave a non-blind robust algorithm. Vertices are scrambled and divided into several groups using a selected secret key and in each of these groups one bit is inserted by modifying the lengths from its member vertices to the gravity center of the mesh. The modulation scheme is a simple additive method with an adaptive intensity obtained by a local geometrical analysis of the mesh. The extraction is also quite simple, since it is sufficient to regroup the vertices and to inverse the additive insertion model. However, a pre-processing step of registration and resampling is necessary to extract the watermark (to ensure a sufficient robustness and to recover the same grouping of the vertices at extraction as during the insertion), and this step makes the algorithm non-blind. In fact, one watermark bit is repeatedly inserted in each member vertex within a group. This redundant insertion and the weighting rule at the extraction are the main reasons for its good robustness. Meanwhile, their method is the first attempt in history to insert a watermark in a global and essential geometric characteristic of a 3D mesh (here the lengths from vertices in a same group to the mesh center). In Benedens's "Vertex Flood Algorithm (VFA)" [20], after grouping vertices according to their distances to the center of a designated triangle, the range of the group interval is then divided into $m = 2^n$ subintervals; all the group vertices' distances to the chosen triangle center are then altered so that the new distances all fall into a certain subinterval that stands for the next n watermark bits.

Recently, researchers have tended to embed watermarks in spherical coordinate systems by using certain kinds of histograms. This approach seems promising because some blind and robust techniques have been devised based on modification of these histograms. Zafeiriou *et al.* [21] first calculate the principal axes of the object and afterwards convert the vertices' coordinates into the spherical system (r, θ, φ), then they divide the vertices into several groups associated with different ranges of θ. The histogram of the prediction errors of vertex radial components is constructed for each group. The prediction is calculated from the vertex 1-ring neighbors by applying a local neighborhood operator. The authors assume a Gaussian distribution of these prediction errors in a group, and embed one watermark bit by modifying the left or right side distribution of the histogram. The basic idea is to alter the histogram one-side variance either on the left or on the right so as to indicate the bit '-1' or the bit '+1', respectively. Similarly, Cho *et al.* [22] construct the histogram of the distances between vertices and mesh gravity center, and then divide this histogram in bins associated with different ranges of this distance. They make the hypothesis of a uniform distribution in each bin. Finally, one bit is inserted by slightly altering either the mean value (see Fig. 5) or the variance of the distribution in each bin. Both algorithms proposed by Zafeiriou *et al.* and Cho *et al.* are robust to common geometric attacks and simplification. However, these methods can suffer from the causality problem because the key parameters during the histogram reconstruction, such as the gravity center in both methods and the principal axes in the method of Zafeiriou *et al.*, could have been changed after watermark embedding. Unfortunately, neither of these two papers has clearly discussed the impact of this problem on the algorithm's robustness. We have concluded from our own experimental results that its impact generally cannot be ignored. Nevertheless, the basic idea of their algorithms deserves deeper investigation because the statistical mesh shape features implied in these histograms are quite robust and can be excellent watermark carriers.

Furthermore, watermark embedding in the spherical coordinate system, especially in the radial component $r_i = \sqrt{x_i^2 + y_i^2 + z_i^2}$, owns additional advantages. We may devise some similarity-transformation-invariant algorithms if the distance component is relative to the mesh center. Moreover, since the com-

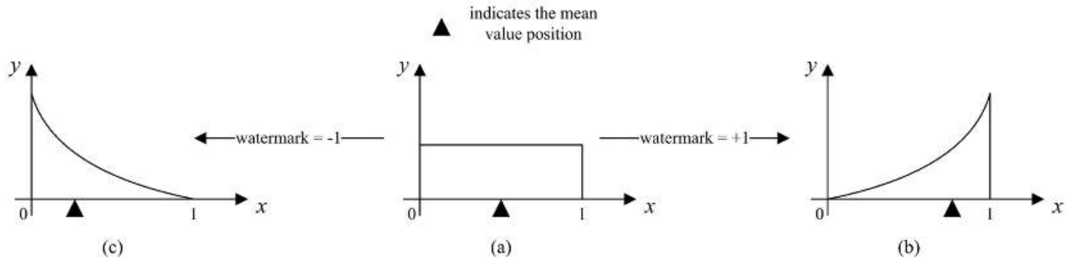

Fig. 5. Watermark embedding in the algorithm of Cho *et al.* [22] that modifies the mean value of the histogram in a bin: (a) the assumed uniform distribution in a bin; (b) the mean value is increased to embed a bit '+1'; (c) the mean value is decreased to embed a bit '-1'. The horizontal axis indicates the normalized distances from vertices to the mesh gravity center (normalized relative vertex norms), and the vertical axis represents the occurrence probability.

ponent r_i represents approximately the mesh shape, its modification is supposed to be more robust than a single x_i, y_i or z_i component modification. These are two other reasons for why numerous researchers chose to watermark in the spherical coordinate system [21]–[23].

Facets have several interesting measures for watermarking. Ohbuchi *et al.* [24] chose the ratio between the height of a triangle and its opposite edge length as primitive to construct a watermarking technique that is intrinsically invariant to similarity transformations (Triangle Similarity Quadruple (TSQ) algorithm). Benedens [20] reported a blind algorithm in which the triangular facet height is quantized. In another Benedens's method [25], the Extended Gaussian Image (EGI) of a mesh is established by clustering facets according to their normal directions, then, in each bin of the EGI, the average normal direction of the group of facets is modified to carry one watermark bit. Since these average normal directions approximately describe the mesh shape, this scheme is demonstrated to be relatively robust to simplification and remeshing. Kwon *et al.* [26] proposed a similar approach based on EGI. Both algorithms are semi-blind because they need to recover the original mesh pose in the 3D space at the extraction to achieve an invariant EGI. Instead of EGI, Lee *et al.* [27] adopted Complex EGI for watermarking. They construct the EGI in the same way, but associate each bin with a complex weight, which depends not only on the bin's total surface size but also on the proximity of the involved facets. In their algorithm, the bins with bigger complex weights are selected as carriers, and this selection is proven to be able to reinforce the robustness. One inconvenience of the facet-based algorithms is that the modification of the positions of the involved vertices is indirect and sometimes quite complicated, especially in the last three algorithms based on EGI or complex EGI. In general, the motivation to embed watermark in facets is mainly to reinforce the robustness, especially to similarity transformation and simplification. However, the final modification on vertices is indirect, and it is sometimes difficult to control the introduced distortion and the expected robustness.

There exist other spatial techniques that modify the geometry, which are not so robust but all have some particularity worth mentioning. Ohbuchi *et al.* [24] presented the "Tetrahedral Volume Ratio Embedding" algorithm that is invariant to affine transformation. Song and Cho [28] provided an interesting means for easily using existing image watermarking techniques. A bounding cylinder is generated from the cover mesh, and then a regular sampling is carried out on the profile of this cylinder. For each sample, the authors calculate the horizontal geodesic distance from the sample to the mesh surface and take this value as the brightness of this sample pixel. A watermark can then be inserted in this pseudo-range image. The changes on horizontal geodesic distances after watermarking have to be reflected on the 3D mesh by modifying the positions of related vertices. At last, Bennour and Dugelay [29] proposed to insert watermarks in the 2D contours of a 3D mesh object.

To summarize, the main drawback of the robust techniques in spatial domain is their relatively weak robustness to connectivity attacks, except the histogram-based and EGI-based techniques. For blind schemes, the synchronization issue is really a difficult problem, because both the attacks and the insertion process itself (causality problem) can desynchronize the watermark. However, these methods often have the advantage of high capacity, and are easy to implement.

2) Robust Techniques in Transform Domain: Most of the successful robust image watermarking algorithms are based on spectral analysis. A better imperceptibility can be gained because it has an information spreading effect of the inserted watermark bits in all the spatial and spectral parts of the cover content. A better robustness can also be achieved if the watermark is inserted in the low and median frequency parts. Unfortunately, for 3D meshes, there does not yet exist an efficient and robust spectral analysis tool. Moreover, the lack of a natural parameterization and the irregular sampling make spectral analysis even more difficult. As it can be seen in the following subsections, almost all the existing spectral analysis tools have their limitations. In addition to the algorithms that embed watermarks in the spectrum obtained by direct frequency analysis, we also present here the algorithms that are based on multiresolution analysis. The basic idea behind both of them is the same: modification of the data obtained after a certain transformation.

- *Robust Techniques in Transform Domain Based on Direct Frequency Analysis*

Researchers have tried different types of frequency analysis, but all of them have their limitations or deficiencies.

For spectral analysis based on a Laplacian matrix, a matrix D of dimension $N \times N$ (N being the number of mesh vertices) is constructed based on mesh connectivity. The construction of this symmetric matrix is quite simple and it implies the adjacency relations between vertices. If the vertices v_i and v_j

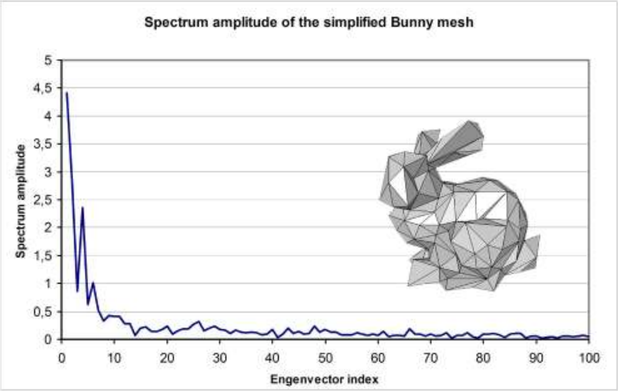

Fig. 6. The spectrum amplitude of the simplified Bunny mesh model (100 vertices).

are connected by an edge, then the elements d_{ij} and d_{ji} of the matrix D are set to -1; otherwise, they are set to 0. Each diagonal element d_{ii} is equal to the valence of the vertex v_i. The N-sized spectral vectors $O = (o_1, o_2, ..., o_N)$, $P = (p_1, p_2, ..., p_N)$, $Q = (q_1, q_2, ..., q_N)$ are calculated respectively as the projections of the three coordinate vectors $X = (x_1, x_2, ..., x_N)$, $Y = (y_1, y_2, ..., y_N)$, $Z = (z_1, z_2, ..., z_N)$ on the N normalized eigenvectors of this Laplacian matrix, which have been sorted in an ascending order according to their associated eigenvalues. Fig. 6 illustrates the spectrum amplitude of the simplified Bunny mesh (100 vertices). The i_{th} spectrum amplitude coefficient is calculated as $s_i = \sqrt{o_i^2 + p_i^2 + q_i^2}$. This mesh spectral analysis tool was originally introduced in graph theory, and then used by Karni and Gotsman [30] for mesh compression. Later, based on this analysis, Ohbuchi *et al.* [31] proposed a non-blind watermarking method (additive modulation of the low and median frequency coefficients), while Cayre *et al.* [32] gave a blind one (quantization of the low and median frequency coefficients).

There exist two serious problems with the Laplacian frequency analysis. First, the computation time increases rapidly with mesh complexity due to the diagonalization of the $N \times N$ Laplacian matrix. This complexity problem forced the authors to cut the original mesh into several patches possessing fewer vertices. Secondly, the analysis procedure depends on the mesh connectivity information and on the vertex ordering. To overcome this fragility to connectivity change and vertex reordering, the authors proposed a pre-processing step of resampling at the extraction phase in order to recover exactly the same connectivity and vertex ordering as in the cover mesh.

Wu and Kobbelt [33] reported another spectral algorithm that is based on radial basis functions. The construction of these basis functions is relative to the geometric information. This kind of analysis seems effective because it can provide a good approximation of the original mesh with just a very limited number of basis functions. Thus, calculation time can be greatly saved. In spite of this improvement, the algorithm remains sensitive to various attacks and dependent on the vertex ordering. For this reason, the authors still proposed to carry out registration and resampling before the real extraction. With the similar objective of solving the computation performance problem, Murotani and Sugihara [34] proposed to watermark the singular spectral coefficients. In this method, the matrix to be diagonalized has a much lower dimension. However, the robustness problem still exists and the algorithm remains non-blind to ensure robustness to connectivity changes.

Although current 3D mesh spectral analysis tools are not efficient and robust enough, they provide the opportunity to make use of the existing mature spectral watermarking techniques of digital images.

- *Robust Techniques in Transform Domain Based on Multiresolution Analysis*

Based on the regular wavelet analysis tool presented in the subsection concerning fragile techniques in transform domain, Kanai *et al.* [35] proposed a non-blind algorithm that modifies the ratio between a wavelet coefficient norm and the length of its support edge, which is invariant to similarity transformations. Uccheddu *et al.* [36] described a blind one-bit watermarking algorithm with the hypothesis of the statistical independence between the wavelet coefficients norms and the inserted watermark bit string. However, in order to obtain the blindness, they also sacrificed the robustness to connectivity attacks.

Thanks to a remeshing step in spatial domain before wavelet decomposition, which is introduced by Eck *et al.* [37], the regular wavelet analysis can be extended to irregular meshes. This remeshing step can also be done in the spherical parameterized space. Jin *et al.* [38] used such a technique to insert a watermark into both the coarsest representation and the spherical wavelet [39] coefficients of an irregular mesh. However, this remeshing step does not seem robust enough, and it can bring some extent of noise to the watermark, which would cause extraction error. Using a direct irregular mesh wavelet analysis tool without any assisting remeshing step [40], Kim *et al.* [41] devised a blind algorithm. However, this method is still fragile to connectivity attacks.

Other multiresolution analysis tools are also employed to develop 3D mesh watermarking algorithms. Hoppe [42] presented a multiresolution decomposition method based on iterative edge collapse operations. The dual reconstruction procedure is based on iterative vertex split operations. Praun *et al.* [43] applied these decomposition and reconstruction methods for watermarking. They picked out the vertex split steps of the reconstruction process that introduced the most significant geometric modifications. For each vertex to be split in these selected steps, they defined a zone containing all its incident facets in the coarse mesh. They then found the corresponding area in the original dense mesh and took this area as the watermark carrier. One bit was inserted in each area by deforming it using a modulation function. Actually, their watermarking technique lies between spatial and classical spectral methods. Here, the multiresolution analysis serves to find the "low frequency", salient, spatial parts of the mesh, and the insertion in these parts is supposed to be more robust. Unfortunately, these iterative edge collapse operations are still dependent on the mesh connectivity. Thus, this algorithm is non-blind mainly due to the connectivity recovery before extraction. At last, Yin *et al.* [44] embedded a robust, but non-blind watermark (connectivity recovery is also necessary) in the coarsest representation after a mesh multiresolution analysis based on the Burt-Adelson pyramid decomposition [45].

Nonetheless, just as in the current direct spectral analysis tools, the available multiresolution analysis schemes have either connectivity restrictions or robustness deficiencies (especially to connectivity attacks). For the majority of these techniques, registration and resampling are recommended to ensure a sufficient robustness; but this pre-processing step inevitably makes the algorithms non-blind.

Besides the direct spectral analysis and the multiresolution analysis, mesh parameterization [46] is also used for watermarking. Parameterization is a technique that transforms a 3D mesh into a bidimensional description, and thus probably enables the use of the existing 2D image watermarking algorithms. Li *et al.* [47] converted the initial mesh in the spherical parameterization domain and watermarked its 2D spherical harmonic coefficients. This algorithm is a semi-blind one since it needs the spherical parameterization information of the original non-watermarked mesh at extraction to ensure a sufficient robustness.

III. Attack-Centric Investigation

As mentioned in subsection II.A, the attacks constitute a critical factor when designing 3D mesh watermarking algorithms. In this section, we carefully discuss three types of attacks and introduce the existing solutions in the literature.

A. Robustness to Geometric Attacks

This kind of attack only modifies the geometric part of the watermarked mesh. No matter what is the nature of the geometric change, the attack is reflected by a modification of vertex positions.

1) Similarity Transformations: Similarity transformation is considered to be a common operation rather than an attack, against which even a fragile watermark should be able to stand. It includes translation, rotation, uniform scaling, and combinations of the above three operations. Generally speaking, there are three different strategies to build a watermark that is immune to these attacks.

The first solution is to use some primitives that are invariant to similarity transformations. Ohbuchi *et al.* [5] provided a list of such primitives. The most utilized is the ratio between two measures of a triangle (height or edge length). Some primitives used in existing blind spatial techniques are also invariant to similarity transformations, like the quantized position of the projection of a vertex on its opposite edge in a triangle [10], and the relative position of a vertex to a zone defined by its 1-ring neighbors [11]. Practically, these primitives are all some relative measures between several absolute and individual ones.

Not only the watermarking primitives, but also the synchronization schemes have to be insensitive to similarity transformation. Existing synchronization mechanisms often consist of criteria for choosing the first primitive and further spreading schemes. For example, in [10], the authors consider every triangle as a two-state object with one entry edge and two exit edges. They take the longest edge in a certain facet intersecting with the mesh most significant principal axis as the first entry edge. The spreading scheme is determined by a secret key: if the next bit in this key is '0', then the first edge in the clockwise direction from the entry edge inside the current facet is chosen as the next entry edge and the next triangle is thus determined, and *vice versa*. In [11], the reference vertex is selected as the one with the smallest average incident edges length, and the vertices to be watermarked are ordered by their distances to this reference vertex. The causality problem arises in the second mechanism because after watermark insertion, the order of the vertices may have been changed. That is why the author introduces a post-processing step to rectify this order. Another option is the so-called indexing scheme. One example is given in [5]. A group of four triangles are combined together as a primitive. One of them is modified to indicate the existence of watermark bits in this macro-group. Two other triangles are used to hide the real watermark bits. The index of these bits in the entire watermark sequence is hidden in the last triangle. The advantage of this option is that the extraction failure of a certain bit (or certain bits) will not influence the extraction (with correct indices) of the posterior bits, but at the same time it decreases the capacity.

The invariance to similarity transformation can be also achieved in a wavelet domain by watermarking the ratio between the norm of a wavelet coefficient and its support edge length [16], [35]. Moreover, if we expect robustness even to affine transformations, the Nielson-Foley norm [48] can be a good primitive candidate. Benedens and Busch [49] quantize this norm, and Wagner [50] replaces some medium-important bits of this norm to insert watermarks.

The second solution is to watermark in an invariant space. One such space can be obtained by carrying out the following steps [51].

1) Translate the origin of the objective Cartesian coordinate system to the mesh gravity center.
2) Carry out a uniform scaling so that the whole mesh is bounded in a unit sphere or cube.
3) Calculate the principal axes of the mesh and reorientate the object so that they coincide with axes of the Cartesian coordinate system.

The watermark is then inserted in this new space. But the causality problem arises because the variables

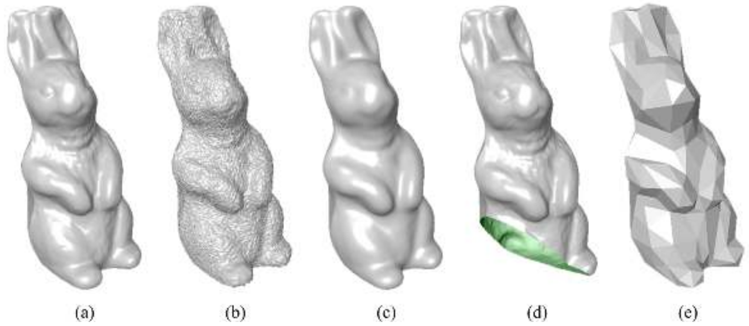

Fig. 7. Original mesh and four examples of attacked meshes: (a) original rabbit mesh; (b) random noise addition; (c) smoothing; (d) cropping; (e) simplification.

used in the above steps, such as the gravity center and the principle axes orientations are probably changed after watermark insertion. Therefore, there will possibly exist some extent of errors when reconstructing this space at the extraction. If a precise extraction is demanded, this introduced error cannot be ignored. Therefore, at least some feature values of the insertion space have to be memorized, but this will make the technique semi-blind, or even non-blind. Note that not all watermark embedding schemes need all of the above three steps: which steps are needed depends on the nature of the watermark primitive.

The third solution is to carry out the registration of the input mesh at extraction with the original non-watermarked one. Low-precision registration methods use singular spectral coefficients [34], eigenvectors of the vertices' correlation matrices [52], inertial moments [31], and characteristic points [33] of the two meshes. High-precision methods often need user interactions to determine a good initial condition, and then the registration is realized by iteratively minimizing a sum of local errors [31], [44]. This solution will obviously make the algorithms non-blind, but provides a better robustness.

2) Signal Processing Attacks: A mesh can be considered as a signal in the three-dimensional space. There are counterparts of the traditional one-dimensional signal processing techniques for 3D meshes, such as random noise addition, smoothing, enhancement, and compression (usually realized by quantization). Fig. 7.b and Fig. 7.c illustrate two examples. Although these operations can be very harmful to inserted watermarks, they are really common manipulations in animation and special effects applications.

Random noise, smoothing, and enhancement can be modeled in the spectral domain by a modification of the high-frequency part. Quantization can be thought as a certain form of noise, but its effect is somewhat complicated. Generally speaking, the transform-domain-based watermarking techniques that modify the low and median frequency parts are more robust to these attacks, as demonstrated by Praun *et al.* [43]. Their method is among the most robust in the literature. Note that for the additive watermarking schemes, which insert the watermark by modulating (*i.e.* perturbing) spectral coefficients obtained by direct frequency analysis, insertion in the low frequency part is both more robust and more imperceptible compared to insertion in the high frequency part if under the same additive embedding intensity. Different additive modulation schemes have been developed. Ohbuchi *et al.* [52] proposed to repeat the watermark insertion in the first half of the spectrum with a constant intensity. Wu and Kobbelt [33] watermarked only the very low frequency coefficients and proposed an adaptive insertion intensity that is proportional to the amplitude of the coefficient. Lavoué *et al.* [53] gave another modulation scheme, in which the intensity is linear for the low and median frequency coefficients and constant for the high frequency part.

Spatial techniques are less robust to signal processing attacks. One exception is the histogram-based techniques [21], [22]. Statistical mesh shape features used in these techniques are rather preserved after such attacks because they represent global descriptors of groups of mesh combinatorial elements. Another efficient solution is to search for an adaptive spatial insertion intensity based on local geometric analysis. This analysis can be based on the average length of the incident edges of a vertex [54], the geometric distortion introduced by a vertex split operation [43], the minimal incident edge length of a vertex [44], or the possible normal direction variance of the incident facets of a vertex after insertion [19]. The basic idea is to increase the watermarking intensity while ensuring the visual quality. At last, redundant insertion [52] and use of error correction codes [53] can sometimes significantly reinforce the robustness to these signal processing attacks.

3) Local Deformation Attacks: A local deformation is sometimes imperceptible if we do not have the original mesh for comparison, but it can seriously disturb the watermark, especially the synchronization process.

One natural solution is to divide the mesh into several patches and repeat the watermark insertion in each patch. This decomposition can be based on curvature or semantic analysis, or simply on a discretization of the θ and/or φ domain in the spherical coordinate system. As mentioned previously, segmentation into patches may also decrease the insertion time for some transform-domain-based techniques. At the extraction, one has to realize exactly the same decomposition. That is relatively simple and robust for non-blind techniques thanks to the availability of the cover mesh or the non-attacked stego-mesh. However, designing a blind algorithm capable of resisting local deformation is a difficult task. The segmentation or discretization methods will probably fail at the extraction phase because the key parameters, such as curvature, mesh gravity center, or principal axes will certainly be disturbed after watermark insertion and a local deformation attack. This situation forces the watermarkers to devise segmentation schemes robust to various attacks, including local deformations. Alface *et al.* [55] have made some efforts on this problem. They carried out a segmentation based on mesh feature points obtained by geodesic distance analysis, which is particularly robust to local deformations and cropping. Whereas, the robustness of their blind watermarking algorithm still seems to need improvement. Another solution for resisting local deformation is the indexing mechanisms, as mentioned in the subsection concerning the robustness against similarity transformations. However, it is not easy to derive an indexing watermark that can withstand the connectivity attacks.

B. Robustness to Connectivity Attacks

This class of attacks includes cropping, remeshing, subdivision and simplification. Usually, they are quite difficult to handle.

Cropping is a special attack (see Fig. 7.d for an example), and some researchers prefer to regard it as a geometric attack because its consequence is quite similar to the one caused by local deformation. Watermark repetition in different patches and indexing schemes seem to be the most efficient ways to resist cropping.

With regard to the other connectivity attacks (Fig. 7.e illustrates an example of simplification), the algorithms that take the average normal directions of groups of facets as primitives [25], [27], or the histogram-based algorithms [21], [22], seem less sensitive. These primitives approximately describe the mesh shape and thus are partly conserved after connectivity modification. Note that although the above histogram-based techniques are robust to mesh simplification, they remain vulnerable to non-uniform remeshing and subdivision. These attacks will seriously modify the distribution of these histograms and cause the failure of the watermark extraction. Other spatial techniques are less robust by reasons of both the geometric change of the primitives and the desynchronization problem. The basis function construction and the frequency coefficients calculation in existing direct spectral analysis tools are either dependent on vertex order [31], [33], [34] or on mesh connectivity [31]. Similarly, the existing multiresolution analysis tools often have connectivity restrictions, or are not robust enough to connectivity changes. Hence, to attain a sufficient robustness for these methods, the authors usually recommend processing connectivity restoration before extraction. This restoration procedure can be considered a resampling of the extraction input mesh (objective mesh) so as to obtain the same connectivity configuration as the cover mesh [19], [31], [44] or the non-attacked stego-mesh [33] (reference mesh). The task is to find, for each vertex in the reference mesh, a corresponding point on the surface of the objective mesh. This correspondence can be established by the nearest neighbor criterion [33], ray intersection [19], [31], or iterations targeting to minimize a particular cost function [44].

Two other possibilities to handle connectivity attacks are to find a robust transformation or parameterization domain that is not sensitive to connectivity change, or to insert watermarks in some robust mesh shape descriptors.

C. Robustness to Other Attacks

This group contains mainly three attacks: file attack, format attack, and representation attack. The file attack simply consists in reordering the vertices and/or the facets in the mesh description file. The mesh file format conversion attack may alter the underlying mesh data structure, so that the intrinsic processing order of the vertices and facets can also be changed. In order to be invariant to these two attacks, one just needs to make the synchronization scheme independent of these intrinsic orders. The representation conversion may be the most destructive attack to 3D mesh watermarks, because after such an attack, the mesh itself will no longer exist (for example, an approximation of a mesh with an NURBS model). Until now, no researcher has mentioned robustness to this attack. In our opinion, the two ideas given at the end of the last subsection can also be potential solutions to resisting this serious attack.

IV. Discussion and Perspectives

Table I and II present a comparison of some typical algorithms of each class. The values in the column "Inserted bits" are the ones reported in the original papers. Most robustness performances are evaluated qualitatively by a sign ranging from '$--$', which means the least robust, to '$++$', which stands for the most robust. In these two tables, the algorithms are classified according to their watermarking primitives. The first four algorithms in the class of "Spatial techniques on vertices" and the first algorithm in the class of "Multiresolution analysis techniques" are fragile ones, and all the other algorithms can be considered as robust techniques. In the class of "Other techniques", we list two other representative algorithms (one in spatial domain, and the other one in transformation domain), which do not belong to any of the other four classes.

In our opinion, there exist many valuable research themes in 3D mesh watermarking research, as listed in the following paragraphs. Some new ideas, which are possibly the potential solutions to these topics, are also presented.

A. Classic Problem: Trade-off between Capacity, Robustness, and Imperceptibility.

These measures are often contradictory. For example, an important watermarking intensity provides a better robustness, but normally degrades the visual quality of the watermarked mesh and risks making the watermark perceptible. The redundant insertion could considerably strengthen the robustness, but meanwhile unavoidably decreases the capacity. Local adaptive geometric analysis seems favorable to find

TABLE I

COMPARISON OF DIFFERENT 3D MESH WATERMARKING ALGORITHMS.

Categories	Algorithms	Clearly controllable intensity	Inserted bits	Blindness	Local adaptability
Spatial techniques on vertices	Yeo and Yeung [7]	No	1 bit/vertex	Yes	No
	Lin *et al.* [8]	Yes	1 bit/vertex	Yes	No
	Cayre and Macq [10]	Yes	1 bit/vertex	Yes	No
	Bors [11]	No	0.2 bits/vertex	Yes	Yes
	Yu *et al.* [19]	Yes	≈50 bits	No	Yes
	VFA [20]	Yes	≈900 bits	Yes	No
	Zafeiriou *et al.* [21]	No	≈20 bits	Yes	No
	Cho *et al.* [22]	Yes	64 bits	Yes	No
Spatial techniques on facets	TSQ [5], [24]	No	1.2 bits/facet	Yes	No
	Benedens [25]	Yes	≈30 bits	Semi	No
	Lee *et al.* [27]	Yes	≈50 bits	Semi	Yes
Direct spectral analysis techniques	Ohbuchi *et al.* [31]	Yes	32 bits	No	No
	Cayre *et al.* [32]	Yes	64 bits	Yes	No
	Wu and Kobbelt [33]	Yes	24 bits	No	No
	Alface and Macq [56]	Yes	64 bits	Yes	No
Multiresolution analysis techniques	Wang *et al.* [17]	Yes	1.5 bits/vertex	Yes	No
	Kanai *et al.* [35]	Yes	≈620 bytes	No	No
	Uccheddu *et al.* [36]	Yes	1 bit	Yes	No
	Praun *et al.* [43]	Yes	50 bits	No	Yes
	Yin *et al.* [44]	Yes	250 bits	No	Yes
Other techniques	Bennour and Dugelay [29]	Yes	≈500 bits	No	No
	Li *et al.* [47]	No	24 bits	Semi	No

TABLE II

CONTINUATION OF TABLE 1: ROBUSTNESS OF DIFFERENT ALGORITHMS TO VARIOUS ATTACKS.

Algorithms	Similarity transformation	Signal processing attacks	Local deformation and cropping	Connectivity attacks	Elements reordering
Yeo and Yeung [7]	−−	−−	Localization*	−−	Fragile
Lin *et al.* [8]	−−	−	Localization*	−−	Invariant
Cayre and Macq [10]	++	−	−−	−−	Invariant
Bors [11]	++	−	−	−−	Invariant
Yu *et al.* [19]	Registration	+	−	Resampling	Invariant
VFA [20]	+	−	−	−	Invariant
Zafeiriou *et al.* [21]	+	+	−	+	Invariant
Cho *et al.* [22]	+	+	−	+	Invariant
TSQ [5], [24]	++	−	+	−−	Invariant
Benedens [25]	Registration	+	−	+	Invariant
Lee *et al.* [27]	Registration	+	−	+	Invariant
Ohbuchi *et al.* [31]	Registration	++	++	Resampling	Resampling
Cayre *et al.* [32]	+	+	++	−−	Fragile
Wu and Kobbelt [33]	Registration	++	++	Resampling	Resampling
Alface and Macq [56]	+	+	++	+	Fragile
Wang *et al.* [17]	++	−−	Localization*	−−	Invariant
Kanai *et al.* [35]	+	−	−	−−	Invariant
Uccheddu *et al.* [36]	−	+	−	−	Invariant
Praun *et al.* [43]	Registration	++	++	Resampling	Resampling
Yin *et al.* [44]	Registration	+	−	Resampling	Resampling
Bennour and Dugelay [29]	Registration	+	+	−	Invariant
Li *et al.* [47]	+	+	+	Resampling	Invariant

*"Localization" means the capability of locating attacks for the fragile algorithms.

optimum watermarking parameters in order to achieve a sufficient compromise between these indicators. A valuable solution could lie in detecting rough (noised) regions where slight geometric distortions would be nearly invisible [11], [57], [58]. As observed in [11], these regions are characterized by the presence of many short edges, and they are somewhat equivalent to highly textured or detailed image areas, which are often used by image watermarking algorithms to obtain a better invisibility.

B. Algorithm Evaluation.

So far, the research community has lacked a widely used performance evaluation system of the existing algorithms. We at least need a standard attack benchmark and distortion measurement.

For the latter subject, Benedens *et al.* [59] first presented a study of different criteria to take into account in order to insure imperceptibility of a watermark. They emphasize the importance of preserving the continuity and the symmetry of the surface. One of the most critical points is the imperceptibility; indeed the visual distortion introduced by the watermark embedding has to be nearly invisible to a human eye. However, classical metrics based on geometric differences like the Hausdorff distance, available in much software [60], [61] do not match well with the human visual perception. Hence, some authors have proposed perceptual distortion measures: Corsini *et al.* [62] introduced some perceptual metrics, based on global roughness variation, to measure the quality of a watermarked mesh. They argue that the presence of visual artefacts produced by the watermark is reflected by the amount of roughness introduced on the surface. They define two distinct roughness measures, which are matched with subjective experiments based on human evaluations. Similarly, based on curvature analysis in local windows of the mesh, Lavoué *et al.* [63] introduced a 3D perceptual metric following the concept of structural similarity. Finally, Alface *et al.* [64] presented two metrics to benchmark watermarking schemes: one was based on a measure of distortion between several 2D views of the 3D objects, and the second was based on the distortion of energy calculated using 2D parameterization. These metrics however, do not incorporate a subjective experiment.

C. Construction of Robust and Blind Algorithms.

The design of such algorithms attracts the attention of many researchers considering their satisfactory flexibility and reliability. In our opinion, this will require overcoming at least two difficulties. The first one is building a robust and secure synchronization mechanism, especially for spatial techniques. As mentioned before, the problem of desynchronization can be caused by both the causality problem during watermark insertion, and the attacks on the watermarked objects after insertion. Using certain robust aspects of the mesh to locate and index the watermarking primitives seems to be a good idea. For example, regarding the watermarking of a semi-regular mesh, we can consider the wavelet coefficients of the coarsest-level mesh as the primitives and index them by using the lengths of their associated edges [18]. This ordering is experimentally very robust to geometric attacks and ensures a robust synchronization. Another advantage is that the causality problem is avoided. After we have modified the norms of the wavelet coefficients to insert the watermark, their indices are not altered. One special difficulty in 3D mesh watermarking is that we often have to establish an ordering of the watermark carriers according to their own properties, which at the same time are the watermark primitives. On the contrary, the locations and brightness (or color) of the pixels in an image are clearly separated, and even after a desynchronization attack, such as a rotation, it seems not very difficult to recover the original order in a blind way.

The second difficulty is avoiding the registration and resampling pre-processing step, which succeeds in ensuring the robustness, but inevitably makes the methods non-blind. As mentioned before, global and robust shape descriptors or transformations, like geometric moments or spherical harmonic transformation, can be good starting points. Some existing blind algorithms make use of a blind registration process at the extraction. This process attempts to rebuild the same watermarking space as during the insertion. It often contains translation [21], [22], reorientation [21], and uniform scaling. In existing algorithms, this blind registration is based only on vertices that just represent a discrete sampling of the real continuous surface, and thus can provide inaccurate results. For example, the coordinates of the mesh gravity center are usually defined as the average coordinates of all the vertices, thus the result will be incorrectly displaced toward the mesh part where the sampling density is higher. This discrete calculation is also vulnerable to noise, smoothing, and of course connectivity changes. One solution is to compute the statistics, on which the blind registration depends, using points sampled on the surface of the mesh with uniform distribution (*i.e.* inside the facets), rather than on the vertex coordinates. Another solution is to process this blind registration in a more precise way, by using the analytic volume or surface moments. Tuzikov *et al.* [65] established the mathematical expression for the calculation of these continuous moments directly from the vertices coordinates. The values of these moments and the final blind registration result have proven to be much more robust than the discrete calculation [66]. The blind registration also suffers from the causality problem as mentioned before. To overcome this problem, one good solution is to separate the mesh elements into two groups: watermark carriers and adjusting elements [9]. After the watermark insertion in the former, the positions of the latter are modified in order to compensate for the influence of the watermark insertion on the blind registration. Fig. 8 illustrates such an example: the θ domain of the spherical coordinate system is discretized and all the vertices are divided into two groups represented by gray and black zones, respectively; the vertices in the gray zones serve to embed the watermark, while

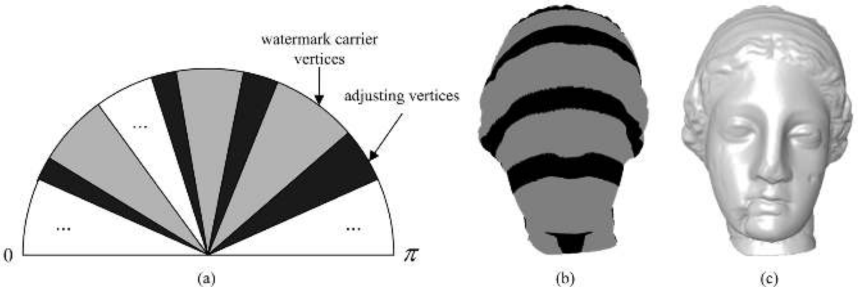

Fig. 8. (a) The θ domain is divided into several intervals with different ranges; (b) for the Venus head mesh, the vertices in gray zones are watermark carriers, and the vertices in black zones serve to process rectification after embedding; (c) the normally rendered Venus head mesh is also illustrated.

the vertices in the black zones play the role of adjusting elements. However, this rectification scheme should be optimized in order to avoid visual distortions, especially on the borders of the zones.

D. Two Ideas toward Stronger Robustness.

Among all types of attacks, the local deformation, cropping, simplification and remeshing are the most intractable ones, especially in the case of blind algorithms. In this subsection, we present two ideas that are possibly effective ways to achieving stronger robustness against these attacks.

1) Remeshing at Both Insertion and Extraction: One possibility to achieve robustness against hard connectivity attacks is to introduce a remeshing step at both the insertion and extraction sides. First of all, the cover mesh (possibly irregular) is remeshed to generate, for instance, a semi-regular mesh with a similar geometrical shape. This procedure is supposed to be composed of two steps: simplification, then subdivision and rectification (*i.e.* vertices displacements). Then, multiple watermarks can be inserted in this semi-regular mesh, which is later taken as the distribution version. For the extraction, the input mesh is also remeshed, and the extraction is executed on the obtained semi-regular mesh. Here, the resistance to connectivity attacks is achieved by introducing a third party (a semi-regular mesh) with identical connectivity at both insertion and extraction, without transmitting any connectivity information (the blindness is thus guaranteed). The key point lies in devising a remeshing scheme that is independent of and insensitive to connectivity changes. Additionally, this semi-regular mesh normally has a negligible geometric distortion compared to the original one and is more favorable for compression thanks to its simple connectivity [67]. Actually, Alface and Macq [56] have made some efforts in this area. They have devised a remeshing scheme based on mesh feature points, which are umbilical points obtained by curvature analysis, but the robustness of their method does not seem strong enough.

Combined with a mesh segmentation scheme based on shape analysis, the above watermarking scheme can also attain robustness against cropping and local deformation. A recent state of the art on 3D mesh segmentation can be found in [68]. However, the main problem lies in constructing a segmentation algorithm that produces similar results when the connectivity changes or when the shape is cropped. We can also envisage an indexing scheme on the semi-regular mesh to resist cropping and local deformation considering that the remeshing step solves the difficulties caused by connectivity attacks.

2) Watermarking with Shape Descriptors: 3D object shape descriptors, usually used for indexing tasks, can be good watermark carrier candidates. Generally speaking, there exist four different groups of descriptors [66]: statistical, transform-based, structural, and multiview-based. The structural descriptors do not seem appropriate for use in watermarking because they are relatively high-level descriptors that are based on the structure or the semantic meaning of the 3D objects. The multiview-based descriptors seem even less adapted.

Practically, the histograms used in [21], [22] are two statistical shape descriptors. They have already been proven to have a satisfying robustness performance, and it is very suitable to construct blind schemes based on them. These two algorithms demonstrate a good robustness to ordinary simplification, but not to non-uniform simplification and remeshing. One possible solution, as mentioned before, is to carry out a uniform resampling of the vertices on the mesh surface before histogram establishment. Another solution is to build a weighted histogram to decrease the contribution of elements from over-sampled areas in the construction of the histogram. For example, the weight of a vertex in the algorithm of Cho *et al.* [22] could be proportional to the total surface of its incident facets. Under the same purpose, a merging step could be introduced to carry out a fusion of several vertices that are very close to each other so as to consider them as only one vertex during the histogram construction.

The next breakpoint may be the transform-based descriptors, which include mainly the geometric moments [69], 3D Fourier transformation [70], 3D Zernike moments [71], 3D angular radial transformation [72], and spherical harmonic transformation [73]. Some of them are particularly interesting because they are intrinsically invariant to rotation and robust to various geometric and connectivity changes. For example, in the case of digital images, 2D moment invariants [74] and 2D Zernike moments [75] have been already used to build geometrically invariant robust and blind watermarks. The transform-based descriptors usually decompose the object into different frequency-like components, so they seem favorable to realize spread spectrum watermarking approaches. Unfortunately, except for surface moments, most of the above descriptors are defined on discretized voxel-based representations. To be able to apply voxel-based descriptors on 3D meshes, we have to first discretize the input mesh into voxels, then after the watermarking insertion, we need some mesh generation technique like the well-known Marching Cubes algorithm [76] to transform the object back into mesh representation. We guess that the noise brought by this last transformation could seriously disturb the inserted watermark and introduce some visible artefacts on the mesh surface. Another option is to generalize the above descriptors for 3D meshes. This generalization seems difficult due to the irregular sampling and the presence of connectivity information. Furthermore, in order to facilitate the watermark insertion, these transformations have to be reversible. If they are not reversible, just like the geometric moments, then we have to realize a time-consuming iterative watermarking process, similar to what is done in [74] in the case of 2D images. Unfortunately, obtaining a reversible transform for 3D mesh seems even more difficult (*i.e.* the mesh object cannot be reconstructed from the descriptors).

E. Other Perspectives.

Other possible research topics include informed 3D mesh watermarking techniques, hierarchical watermarks, 3D mesh digital fingerprints, content-based watermarking, and the interplay between compression and watermarking, or between subdivision and watermarking.

V. Conclusion

Three-dimensional mesh watermarking appears as an interesting and promising research area. We can imagine many potential practical applications of 3D model/graphics watermarking. For example, an automobile constructor could insert watermarks in the car parts it has designed to protect its intellectual properties; a doctor could hide a patient's personal information in the 3D mesh model obtained after a scan, without impacting his diagnosis, to avoid mismatching patient's personal information and his scan result; a mesh data receiver could authenticate the integrity and originality of the mesh model he/she has bought or obtained; even the texture of a mesh model, or the motion parameter of a mesh sequence could be inserted in the mesh description file via watermarking, just like hiding the audio signal of a video within the visual part of the video stream.

However, due to many difficulties stated in section II.A, such as the irregularity of the mesh description and the complexity of the possible attacks, the research work on 3D mesh watermarking is still in its infancy, even after ten years of studies by many contributors. For fragile techniques of arbitrary meshes, constructing an algorithm capable of accurately locating the endured attacks and capable of surviving similarity transformations and vertex reordering is a difficult task. For robust techniques, the causality problem, the desynchronization problem and the attacks (especially the connectivity attacks) are also not so easy to handle. We have provided some working directions to devising robust and blind algorithms. Nearly all of them rely on a supposed efficient analysis or description tool of 3D meshes. They include global or local mesh shape descriptors, robust mesh transformations, and remeshing algorithms insensitive to various attacks. Thus, in our opinion, the most important, also the most difficult part of a 3D mesh watermarking system is the selection of a suitable feature space, in which the watermark signal is inserted. In order to achieve this target, the watermarkers probably should work closely with computer graphics and geometry processing experts.

References

[1] S. Katzenbeisser and F. A. Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*. Artech House Inc., 2000.

[2] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital Watermarking*. Morgan Kaufmann Publishers Inc., 2001.

[3] M. Barni and F. Bartolini, *Watermarking Systems Engineering: Enabling Digital Assets Security and other Applications*. Marcel Dekker Inc., 2004.

[4] K. Wang, G. Lavoué, F. Denis, and A. Baskurt, "Three-dimensional meshes watermarking: Review and attack-centric investigation," in *Proc. of the International Workshop on Information Hiding'07*, 2007, pp. 50–64.

[5] R. Ohbuchi, H. Masuda, and M. Aono, "Watermarking three-dimensional polygonal models," in *Proc. of the ACM International Multimedia Conference and Exhibition'97*, 1997, pp. 261–272.

[6] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.

[7] B. Yeo and M. M. Yeung, "Watermarking 3D objects for verification," *IEEE Computer Graphics and Applications*, vol. 19, no. 1, pp. 36–45, Jan.–Feb. 1999.

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

[9] C. M. Chou and D. C. Tseng, "A public fragile watermarking scheme for 3D model authentication," *Computer-Aided Design*, vol. 38, no. 11, pp. 1154–1165, Nov. 2006.

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

[11] A. G. Bors, "Watermarking mesh-based representations of 3-D objects using local moments," *IEEE Transactions on Image Processing*, vol. 15, no. 3, pp. 687–701, Mar. 2006.

[12] Y. M. Cheng and C. M. Wang, "A high-capacity steganographic approach for 3D polygonal meshes," *The Visual Computer*, vol. 22, no. 9–11, pp. 845–855, Sep. 2006.

[13] H. T. Wu and Y. M. Chueng, "A fragile watermarking scheme for 3D meshes," in *Proc. of the ACM Multimedia and Security Workshop'05*, 2005, pp. 117–124.

[14] N. A. Dodgson, M. S. Floater, and M. A. Sabin, *Advances in Multiresolution for Geometric Modelling*. Springer-Verlag, 2005.

[15] M. Lounsbery, T. D. DeRose, and J. Warren, "Multiresolution analysis for surfaces of arbitrary topological type," *ACM Transactions on Graphics*, vol. 16, no. 1, pp. 34–73, Jan. 1997.

[16] W. H. Cho, M. E. Lee, H. Lim, and S. Y. Park, "Watermarking technique for authentication of 3-D polygonal meshes," in *Proc. of the International Workshop on Digital Watermarking'05*, 2005, pp. 259–270.

[17] K. Wang, G. Lavoué, F. Denis, and A. Baskurt, "A fragile watermarking scheme for authentication of semi-regular meshes," in *Proc. of the Eurographics Short Papers'08*, 2008, (to be published).

[18] ——, “Hierarchical blind watermarking of 3D triangular meshes,” in *Proc. of the IEEE International Conference on Multimedia and Expo’07*, 2007, pp. 1235–1238.

[19] Z. Yu, H. H. S. Ip, and L. F. Kwok, “A robust watermarking scheme for 3D triangular mesh models,” *Pattern Recognition*, vol. 36, no. 11, pp. 2603–2614, Nov. 2003.

[20] O. Benedens, “Two high capacity methods for embedding public watermarks into 3D polygonal models,” in *Proc. of the Multimedia and Security Workshop at ACM Multimedia’99*, 1999, pp. 95–99.

[21] S. Zafeiriou, A. Tefas, and I. Pitas, “Blind robust watermarking schemes for copyright protection of 3D mesh objects,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, no. 5, pp. 596–607, Sep.–Oct. 2005.

[22] J. W. Cho, R. Prost, and H. Y. Jung, “An oblivious watermarking for 3-D polygonal meshes using distribution of vertex norms,” *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 142–155, Jan. 2007.

[23] Y. Maret and T. Ebrahimi, “Data hiding on 3D polygonal meshes,” in *Proc. of the ACM Multimedia and Security Workshop’04*, 2004, pp. 68–74.

[24] R. Ohbuchi, H. Masuda, and M. Aono, “Data embedding algorithms for geometrical and non-geometrical targets in three-dimensional polygonal models,” *Computer Communications*, vol. 21, no. 15, pp. 1344–1354, Oct. 1998.

[25] O. Benedens, “Geometry-based watermarking of 3D models,” *IEEE Computer Graphics and Applications*, vol. 19, no. 1, pp. 46–55, Jan.–Feb. 1999.

[26] K. R. Kwon, S. G. Kwon, S. H. Lee, T. S. Kim, and K. I. Lee, “Watermarking for 3D polygonal meshes using normal vector distributions of each patch,” in *Proc. of the IEEE International Conference on Image Processing’03*, vol. 2, 2003, pp. 499–502.

[27] J. W. Lee, S. H. Lee, K. R. Kwon, and K. I. Lee, “Complex EGI based 3D-mesh watermarking,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E88, no. 6, pp. 1512–1519, Jun. 2005.

[28] H. S. Song and N. I. Cho, “Digital watermarking of 3D geometry,” in *Proc. of the International Symposium on Intelligent Signal Processing and Communication Systems’04*, 2004, pp. 272–277.

[29] J. Bennour and J. L. Dugelay, “Protection of 3D object visual representations,” in *Proc. of the IEEE International Conference on Multimedia and Expo’06*, 2006, pp. 1113–1116.

[30] Z. Karni and C. Gotsman, “Spectral compression of mesh geometry,” in *Proc. the ACM SIGGRAPH Conference on Computer Graphics’00*, 2000, pp. 279–286.

[31] R. Ohbuchi, A. Mukaiyama, and S. Takahashi, “A frequency-domain approach to watermarking 3D shapes,” *Computer Graphics Forum*, vol. 21, no. 3, pp. 373–382, Sep. 2002.

[32] F. Cayre, P. R. Alface, F. Schmitt, B. Macq, and H. Maître, “Application of spectral decomposition to compression and watermarking of 3D triangle mesh geometry,” *Signal Processing: Image Communication*, vol. 18, no. 4, pp. 309–319, Apr. 2003.

[33] J. Wu and L. P. Kobbelt, “Efficient spectral watermarking of large meshes with orthogonal basis functions,” *The Visual Computer*, vol. 21, no. 8–10, pp. 848–857, Sep. 2005.

[34] K. Murotani and K. Sugihara, “Watermarking 3D polygonal meshes using the singular spectrum analysis,” in *Proc. of the IMA International Conference on the Mathematics of Surfaces’03*, 2003, pp. 85–98.

[35] S. Kanai, H. Date, and T. Kishinami, “Digital watermarking for 3D polygons using multiresolution wavelet decomposition,” in *Proc. of the International Workshop on Geometric Modeling: Fundamentals and Applications’98*, 1998, pp. 296–307.

[36] F. Uccheddu, M. Corsini, and M. Barni, “Wavelet-based blind watermarking of 3D models,” in *Proc. of the ACM Multimedia and Security Workshop’04*, 2004, pp. 143–154.

[37] M. Eck, T. D. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle, “Multiresolution analysis of arbitrary meshes,” in *Proc. of the ACM SIGGRAPH Conference on Computer Graphics’95*, 1995, pp. 173–180.

[38] J. Q. Jin, M. Y. Dai, H. J. Bao, and Q. S. Peng, “Watermarking on 3D mesh based on spherical wavelet transform,” *Journal of Zhejiang University: Science*, vol. 5, no. 3, pp. 251–258, Mar. 2004.

[39] P. Schröder and W. Sweldens, “Spherical wavelets: Efficiently representing functions on the sphere,” in *Proc. of the ACM SIGGRAPH Conference on Computer Graphics’95*, 1995, pp. 161–172.

[40] S. Valette and R. Prost, “Wavelet-based multiresolution analysis of irregular surface meshes,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, no. 2, pp. 113–122, Mar.-Apr. 2004.

[41] M. S. Kim, S. Valette, H. Y. Jung, and R. Prost, “Watermarking of 3D irregular meshes based on wavelet multiresolution analysis,” in *Proc. of the International Workshop on Digital Watermarking’05*, 2005, pp. 313–324.

[42] H. Hoppe, “Progressive mesh,” in *Proc. of the ACM SIGGRAPH Conference on Computer Graphics’96*, 1996, pp. 99–108.

[43] E. Praun, H. Hoppe, and A. Finkelstein, “Robust mesh watermarking,” in *Proc. of the ACM SIGGRAPH Conference on Computer Graphics’99*, 1999, pp. 49–56.

[44] K. Yin, Z. Pan, J. Shi, and D. Zhang, “Robust mesh watermarking based on multiresolution processing,” *Computers and Graphics*, vol. 25, no. 3, pp. 409–420, Jun. 2001.

[45] I. Guskov, W. Sweldens, and P. Schröder, “Multiresolution signal processing for meshes,” in *Proc. of the ACM SIGGRAPH Conference on Computer Graphics’99*, 1999, pp. 325–334.

[46] M. S. Floater and K. Hormann, *Advances in Multiresolution for Geometric Modelling*. Springer-Verlag, 2005, ch. Surface parameterization: a tutorial and survey, pp. 157–186.

[47] L. Li, D. Zhang, Z. Pan, J. Shi, K. Zhou, and K. Ye, “Watermarking 3D mesh by spherical parameterization,” *Computers and Graphics*, vol. 28, no. 6, pp. 981–989, Dec. 2004.

[48] G. M. Nielson and T. A. Foley, *Mathematical Methods in Computer Aided Geometric Design*. Academic Press, 1989, ch. A survey of applications of an affine invariant norm, pp. 445–467.

[49] O. Benedens and C. Busch, “Towards blind detection of robust watermarks in polygonal models,” *Computer Graphics Forum*, vol. 19, no. 3, pp. C199–C208, Aug. 2000.

[50] M. G. Wagner, “Robust watermarking of polygonal meshes,” in *Proc. of the Geometric Modeling and Processing’00*, 2000, pp. 201–208.

[51] A. Kalivas, A. Tefas, and I. Pitas, “Watermarking of 3D models using principal component analysis,” in *Proc. of the IEEE International Conference on Acoustic, Speech, and Signal Processing’03*, vol. 1, 2003, pp. 637–640.

[52] R. Ohbuchi, S. Takahashi, T. Miyazawa, and A. Mukaiyama, “Watermarking 3D polygonal meshes in the mesh spectral domain,” in *Proc. of the Graphics Interface’01*, 2001, pp. 9–17.

[53] G. Lavoué, F. Denis, and F. Dupont, “Subdivision surface watermarking,” *Computers and Graphics*, vol. 31, no. 3, pp. 480–492, Jun. 2007.

[54] M. Ashourian and R. Enteshary, “A new masking method for spatial domain watermarking of three-dimensional triangle meshes,” in *Proc. of the IEEE Conference on Convergent Technologies for Asia-Pacific Region’03*, vol. 1, 2003, pp. 428–431.

[55] P. R. Alface, B. Macq, and F. Cayre, “Blind and robust watermarking of 3D models: How to withstand the cropping attack?” in *Proc. of the IEEE International Conference on Image Processing’07*, vol. 5, 2007, pp. 465–468.

[56] P. R. Alface and B. Macq, “Blind watermarking of 3D meshes using robust feature points detection,” in *Proc. of the IEEE International Conference on Image Processing’05*, vol. 1, 2005, pp. 693–696.

[57] F. Uccheddu, M. Corsini, M. Barni, and V. Cappellini, “A roughness-based algorithm for perceptual watermarking of 3D meshes,” in *Proc. of the International Conference on Virtual System and Multimedia’04*, 2004, pp. 934–943.

[58] G. Lavoué, “A roughness measure for 3D mesh visual masking,” in *Proc. of the ACM SIGGRAPH Symposium on Applied Perception in Graphics and Visualization’07*, 2007, pp. 57–60.

[59] O. Benedens, J. Dittmann, and F. A. Petitcolas, “3D watermarking design evaluation,” in *Proc. of the SPIE - The International Society for Optical Engineering*, vol. 5020, 2003, pp. 337–348.

[60] N. Aspert, D. Santa-Cruz, and T. Ebrahimi, “MESH: Measuring error between surfaces using the Hausdorff distance,” in *Proc. of the IEEE International Conference on Multimedia and Expo’02*, 2002, pp. 705–708.

[61] P. Cignoni, C. Rocchini, and R. Scorpigno, “Metro: Measuring error on simplified surfaces,” *Computer Graphics Forum*, vol. 17, no. 2, pp. 167–174, Jun. 1998.

[62] M. Corsini, E. D. Gelasca, T. Ebrahimi, and M. Barni, “Watermarked 3-D mesh quality assessment,” *IEEE Transactions on Multimedia*, vol. 9, no. 2, pp. 247–255, Feb. 2007.

[63] G. Lavoué, E. D. Gelasca, F. Dupont, A. Baskurt, and T. Ebrahimi, “Perceptually driven 3D distance metrics with application to watermarking,” in *Proc. of the SPIE-IS and T Electronic Imaging’06*, vol. 6312, 2006, p. 63120L.

[64] P. R. Alface, M. D. Craene, and B. Macq, “Three-dimensional image quality measurement for the benchmarking of 3D watermarking schemes,” in *Proc. of the SPIE-IS and T Electronic Imaging’05*, vol. 5681, 2005, pp. 230–240.

[65] A. V. Tuzikov, S. A. Sheynin, and P. V. Vasiliev, “Computation of volume and surface body moments,” *Pattern Recognition*, vol. 36, no. 11, pp. 2521–2529, Nov. 2003.

[66] J. Ricard, “3D-object indexing and retrieval, from 2D and 3D queries,” PhD Dissertation, Institut National des Sciences Appliquées de Lyon, France, Dec. 2005.

[67] A. Khodakovsky, P. Schröder, and W. Sweldens, “Progressive geometry compression,” in *Proc. of the ACM SIGGRAPH Conference on Computer Graphics’00*, 2000, pp. 271–278.

[68] M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, and A. Tal, “Mesh segmentation: a comparative study,” in *Proc. of the Shape Modeling International’06*, 2006, pp. 14–25.

[69] D. Saupe and D. V. Vranic, “3D model retrieval with spherical harmonics and moments,” in *Proc. of the Symposium on Pattern Recognition’01*, 2001, pp. 392–397.

[70] D. V. Vranic and D. Saupe, “3D shape descriptor based on 3D Fourier transform,” in *Proc. of the EURASIP Conference on Digital Signal Processing for Multimedia Communications and Services’01*, 2001, pp. 271–274.

[71] M. Novotni and R. Klein, “Shape retrieval using 3D Zernike descriptors,” *Computer-Aided Design*, vol. 36, no. 11, pp. 1047–1062, Sep. 2004.

[72] J. Ricard, D. Coeurjolly, and A. Baskurt, “Generalizations of angular radial transform for 2D and 3D shape retrieval,” *Pattern Recognition Letters*, vol. 26, no. 14, pp. 2174–2186, Oct. 2005.

[73] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs, “A search engine for 3D models,” *ACM Transactions on Graphics*, vol. 22, no. 1, pp. 83–105, Jan. 2003.

[74] M. Alghoniemy and A. H. Tewfik, “Geometric invariance in image watermarking,” *IEEE Transactions on Image Processing*, vol. 13, no. 2, pp. 145–153, Feb. 2004.

[75] H. S. Kim and H. K. Lee, “Invariant image watermark using Zernike moments,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 766–775, Aug. 2003.

[76] W. E. Lorensen and H. E. Cline, “Marching cubes: a high resolution 3D surface construction algorithm,” in *Proc. of the ACM SIGGRAPH Conference on Computer Graphics’87*, vol. 21, 1987, pp. 163–170.