Technical Section

Subdivision surface watermarking

Guillaume Lavoué*, Florence Denis, Florent Dupont

LIRIS UMR 5205, INSA-Lyon and Université Lyon 1, F-69621, France

Abstract

This paper presents a robust non-blind watermarking scheme for subdivision surfaces. The algorithm works in the frequency domain, by modulating spectral coefficients of the subdivision control mesh. The compactness of the watermarking support (a coarse control mesh) has led us to optimize the trade-off between watermarking redundancy (which ensures robustness) and imperceptibility by introducing two contributions: (1) spectral coefficients are perturbed according to a new modulation scheme analysing the spectrum shape and (2) the redundancy is optimized by using error correcting codes coming from telecommunication theory. Since the watermarked surface can be attacked in a subdivided version, we have introduced an algorithm to retrieve the control polyhedron, starting from a subdivided, attacked version. Experiments have shown the high robustness of our scheme against geometry attacks such as noise addition, quantization or non-uniform scaling and also connectivity alterations such as remeshing or simplification.
© 2007 Elsevier Ltd. All rights reserved.

1. Introduction

Watermarking provides a mechanism for copyright protection or ownership assertion of digital media by embedding information in the data. A watermark is associated with different characteristics, depending on its purpose. For copyright protection, the watermark has to be *robust* to survive (i.e. remain detectable) through malicious attacks; on the contrary, for applications like integrity verification, the watermark has rather to be *fragile* to detect any change in the document. An other characteristic of a watermarking algorithm concerns the mark extraction which can be *blind* (the original document is not required to extract the mark) or *non-blind* (the original document is needed).

The last important attribute of a watermark is the *imperceptibility*; indeed, the watermarked document has to be visually near identical to the original. More information about digital watermarking can be found in [1].

There still exist few watermarking algorithms for 3D models, moreover, most of the existing methods concern polygonal meshes and ignore other 3D surface representations and particularly subdivision surfaces. A subdivision surface is a smooth surface defined as the limit surface generated by an infinite number of refinement operations using a subdivision rule on an input coarse control mesh. Hence, it can model a smooth surface of arbitrary topology while keeping a compact storage and a simple representation. Subdivision surfaces are now widely used in computer graphics and have been integrated to the MPEG4 standard [2].

In this context we present a robust, imperceptible, non-blind watermarking scheme for subdivision surfaces to serve ownership claims. The algorithm is based on a frequency domain decomposition of the subdivision control mesh and on spectral coefficient modulation. In order to adapt our algorithm to the compactness of the cover object (the coarse control mesh), we have optimized the trade-off between watermarking redundancy (which ensures robustness) and imperceptibility by introducing a new modulation scheme and error correcting codes (ECC). A so-called *synchronization* process was also introduced to

*Corresponding author. Tel.: +33 04 72 44 83 95;
fax: +33 04 72 43 15 36.

E-mail address: glavoue@liris.cnrs.fr (G. Lavoué).

ensure robustness to attacks against a subdivided version of the surface.

Section 2 presents subdivision surfaces, a state of the art about 3D watermarking and the overview of our framework. Section 3 details our different contributions and the complete watermarking algorithm, while Section 4 gives some results and comparisons with existing methods.

2. Context and overview

2.1. Subdivision surface presentation

A subdivision surface is a smooth (or piecewise smooth) surface defined as the limit surface generated by an infinite number of refinement operations using a subdivision rule on an input coarse control mesh. Hence, it can model a smooth surface of arbitrary topology (contrary to the NURBS model which needs a parametric domain) while keeping a compact storage and a simple representation (a polygonal mesh). Moreover it can be easily displayed to any resolution. Today, many subdivision schemes have been developed, based on quadrilateral [3,4], triangular meshes [5] or both [6]. Moreover special rules have been introduced by Hoppe et al. [7] to handle sharp edges. Fig. 1 shows an example of subdivision surface (Catmull–Clark rules). At each iteration, the base mesh is linearly subdivided and smoothed. Subdivision surfaces have been integrated to the MPEG4 standard [2]. Moreover, a lot of algorithms exist to convert a 3D mesh into a subdivision surface [7–13], particularly because this model is much more compact, in terms of amount of data, than a dense polygonal mesh.

2.2. State of the art on 3D watermarking

There still exist few watermarking methods for 3D models compared with the amount of algorithms available for traditional media such as audio, image and video. Most of the existing methods concern polygonal meshes and ignore other 3D surface representations. To our knowledge, there do not exist watermarking schemes for subdivision surfaces and quite few authors have investigated NURBS surface watermarking: Ohbuchi et al. [14] embed the mark into the knot equations by knot reparameterization, while Lee et al. [15] create a virtual 2D image by sampling the parametric support of the NURBS surface and then apply 2D image watermarking techniques.

Existing techniques concerning 3D meshes can be classified into two main categories, depending if the watermark is embedded in the *spatial* domain (by modifying the geometry or the connectivity) or in the *spectral* domain (by modifying kinds of spectral coefficients).

Spatial techniques: The first watermarking techniques have concerned the spatial domain and were introduced by Ohbuchi et al. [16,17]. They apply topological modifications by subdividing triangles to produce recognizable patterns. They also propose to perturb vertices coordinates to obtain certain desired ratio for some tetrahedra volumes or triangles heights. Yeo and Yeung [18] and more recently Cayre and Macq [19] follow a similar approach for fragile watermarking. In a different way, Benedens et al. [20,21] modify surface normals, in order to increase the robustness to simplification. Finally, Yu et al. [22] perturb the length of the vectors linking the surface vertices to the centre of the 3D object. Although having the benefit of being quite fast and simple to implement, these *spatial* methods do not yet provide enough robustness with respect to some ordinary attacks like noise addition, and are rather adapted for blind fragile watermarking or steganography, like the recent algorithm from Maret and Ebrahimi [23] which considers a similarity invariant space to embed the mark, or Zafeiriou et al. [24].

Spectral techniques: These algorithms decompose the target 3D object into a spectral-like domain, in order to embed the watermark following some signal processing approaches like *spread spectrum*, by modifying spectral coefficients. The first authors to consider such an approach were Kanai et al. [25], who decomposed the mesh by applying the lazy wavelets introduced by Lounsbery et al. [26]. Their algorithm was recently extended to blind detection by Uccheddu et al. [27]. Unfortunately these approaches require the mesh to have a semi-regular subdivision connectivity. Thus, recently, Kim et al. [28] present a similar approach based on irregular wavelet analysis which allows to process arbitrary irregular triangle meshes. Other authors use multiresolution decomposition to decompose the object in a pseudo-spectral way. Praun et al. [29] consider iterative edge collapse operations to construct the multiresolution hierarchy, similar to the *progressive mesh* technique from Hoppe [30]. With the same idea, Yin et al. [31] consider the multiresolution
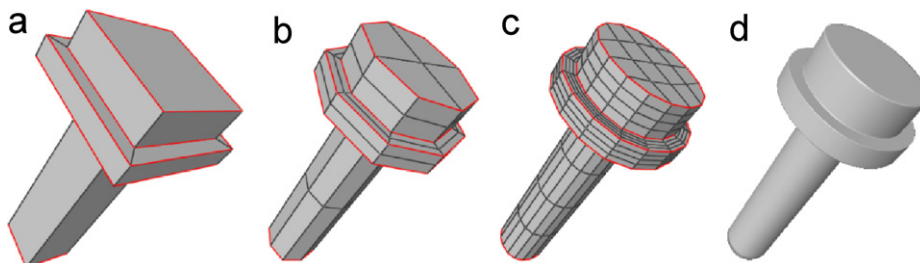

Fig. 1. Example of subdivision surface with sharp edges (in red). (a) Control mesh, (b,c) 1 and 2 subdivision steps, (d) limit surface.

decomposition scheme from Guskov et al. [32]. Finally, Ohbuchi et al. [33,34] employ the spectral mesh analysis proposed by Karni and Gotsman [35]. The mesh is decomposed on the eigenvectors of its Laplacian matrix, which reflect a real spectral decomposition, particularly adapted for watermarking. Unfortunately this decomposition requires a high computation time, which has led the authors to cut the input mesh into several parts before processing. Thus, Wu and Kobbelt [36] have introduced a new set of orthogonal basis functions derived from radial basis functions, allowing to process large meshes. Finally, Li et al. [37] map the input mesh into a sphere (spherical parameterization) and then apply the spherical harmonic transform which provides a kind of Fourier frequency representation of the mesh. Although some blind algorithms exist in these spectral domains [38], most of the spectral techniques presented in this paragraph are additive and not blind, besides the watermarks are mostly embedded in the *low frequencies* [29,31,34,36], in order to minimize the visual distortion and also to remain robust to high-frequency perturbations like noise addition or smoothing. These spectral algorithms are particularly robust to a large variety of attacks such as noise addition, cropping, filtering, simplification, resampling and similarities.

2.3. Objective and framework

Our objective is to propose an efficient watermarking algorithm for subdivision surfaces, which have not been, for the moment, considered in existing 3D techniques, in spite of their popularity and widespread use. Basically, every existing polygonal mesh watermarking technique could be applied on subdivision surfaces since corresponding control polyhedrons *are* polygonal meshes. However, these surfaces have two specificities which cannot be ignored to design a real efficient applicable watermarking scheme:

(1) For a given 3D shape, this representation is much more compact than a polygonal mesh, since the subdivision control polyhedron contains much fewer vertices. Thus there is much less available space to embed the watermark.
(2) Concerning the possible attacks against the watermarked subdivision surface, they can occur on two different states: against the control polyhedron or against a subdivided version.

Taking into account these characteristics, our framework for subdivision surface watermarking, detailed in Fig. 2, is the following:

Our principal objective is the robustness of the mark, thus we have chosen a spectral domain to embed the watermark; among existing decomposition schemes, the spectral analysis from Karni and Gotsman [35] leads to the best decorrelation really close to a theoretical Fourier
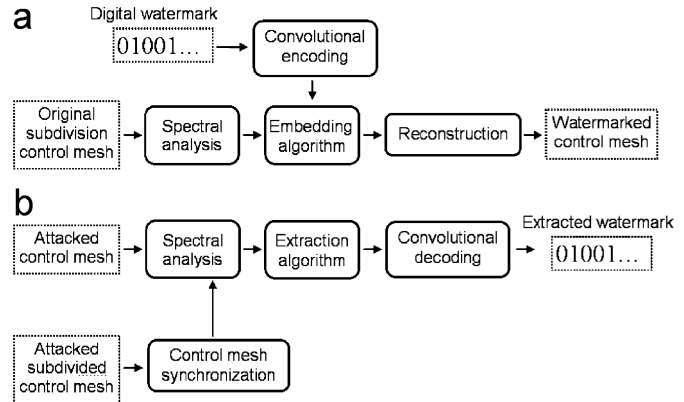

Fig. 2. Our subdivision surface watermarking framework. (a) Watermark embedding, (b) Watermark extraction.

analysis (see Section 3.1). The fact that this decomposition scheme is heavy in calculation is not a problem in our case since we apply it on subdivision control polyhedrons. The compactness of the watermarking support (a coarse control polyhedron) has led us to optimize the efficiency of the insertion in two different ways:

- We propose an extension of the simple additive watermarking scheme, used by most of the authors and particularly by Ohbuchi et al. [33,34], by increasing embedding strength on low-frequency components, in which alterations are less visible for human eyes (see Section 3.2). At the opposite to most of the existing methods, our algorithm will also watermark some high-frequency components, since disturbing a high-frequency of a subdivision control polyhedron has finally a low-frequency impact on the limit surface and therefore leads to low visual distortions.
- In [33,34], the mark is repeated several times to increase the robustness; at the extraction, the extracted marks are averaged to calculate the correlation. We have investigated a more sophisticated technique, coming from telecommunication theory, to increase the robustness of our mark using convolutional encoding (see Section 3.3).

Our extraction process needs to compare the watermarked subdivision control polyhedron with the original one. However, attacks can occur on a subdivided version of the watermarked surface. Thus, we propose an algorithm to retrieve the control polyhedron, starting from a subdivided, attacked (by noise addition, remeshing, simplification) version: the control mesh *synchronization* (see Section 3.4).

3. Subdivision surface watermarking algorithm

3.1. Spectral analysis

The mesh spectrum is obtained by projecting the vertex coordinates on the eigenvalues of the *Laplacian matrix* of

the input polygonal mesh. Karni and Gotsman [35] and Bollabás [39] propose two distinct definitions for the computation of such a matrix. We consider Bollabás's one, which leads to an easier eigenvalues decomposition. The Laplacian matrix L is defined by

$$L = D - A, \tag{1}$$

where D is a diagonal matrix whose each diagonal element d_{ii} corresponds to the valence of the vertex i (the valence is equal to the number of edges connected to this vertex) and A is the adjacency matrix of the mesh whose each element a_{ij} is defined by

$$a_{ij} = \begin{cases} 1 & \text{if vertices } i \text{ and } j \text{ are adjacent,} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

For a mesh with n vertices, matrices A, D and L have an $n \times n$ size. The eigenvalues decomposition of the Laplacian matrix L gives n eigenvalues λ_i and n eigenvectors w_i. By sorting the eigenvalues in an ascending order, the n corresponding eigenvectors form a set of basis functions with increasing frequencies, only depending on the mesh connectivity (geometry is not taken into account). We call W the $n \times n$ projection matrix constructed with the juxtaposition of the n ordered column eigenvectors.

The geometry information of the mesh, containing n vertices $v_i = (x_i, y_i, z_i)$, can be represented by three vectors X, Y and Z:

$$\begin{aligned} X &= (x_1, x_2, \ldots, x_n), \\ Y &= (y_1, y_2, \ldots, y_n), \\ Z &= (z_1, z_2, \ldots, z_n). \end{aligned} \tag{3}$$

The spectral decomposition is obtained by projecting these three vectors on the eigenvector basis and produces three spectral coefficient vectors P, Q and R. These ordered coefficient vectors form three mesh spectra corresponding to the three orthogonal coordinate axes in the spectral domain:

$$\begin{cases} P = W.X, \\ Q = W.Y, \\ R = W.Z. \end{cases} \tag{4}$$

The geometry can be retrieved using spectral coordinates and inverse matrix W^{-1}:

$$\begin{cases} X = W^{-1}.P, \\ Y = W^{-1}.Q, \\ Z = W^{-1}.R. \end{cases} \tag{5}$$

The amplitude spectrum can be obtained by computing coefficients s_i for each vertex by using the transformed coordinates (p_i, q_i, r_i) with the following equation:

$$s_i = \sqrt{(p_i^2 + q_i^2 + r_i^2)}. \tag{6}$$

Fig. 3 presents the amplitude spectrum obtained for the SubRabbit model (200 vertices) which shows a very fast decrease, since most of the geometric information is
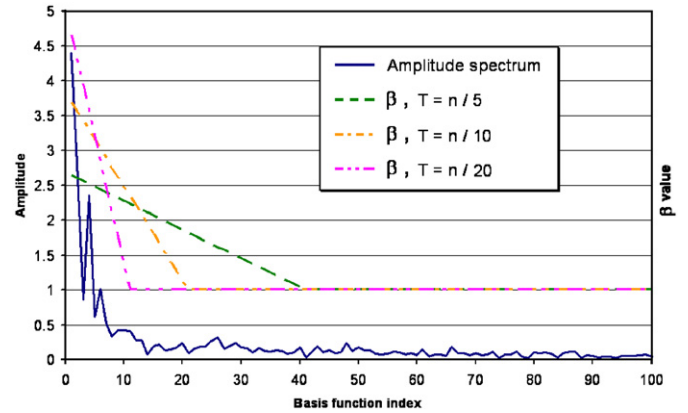

Fig. 3. Amplitude spectrum of the 3D object *SubRabbit* and evolution of the watermarking strength (β function) according to parameter T.

concentrated in low frequencies. We have not represented the first coefficient which corresponds to the continuous component (i.e. the position) of the object and is not considered in the watermarking process.

3.2. Spectral coefficient modulation

Our watermarking algorithm embeds the marks by perturbing the amplitude of the coefficients of the mesh spectra P, Q and R, following the spread-spectrum approach introduced by Cox et al. [40] for 2D image watermarking. For a given modulating vector $V = (v_1, v_2, \ldots, v_m), v_i \in \{-1, 1\}$, there exist several schemes to perturb spectral coefficients, introduced notably by Ohbuchi et al. [33,34] and Wu and Kobbelt [36]. Ohbuchi et al. consider a simple additive scheme

$$\hat{c}_i = c_i + v_i \alpha, \tag{7}$$

with $\hat{c}_i$ the watermarked spectral coefficient, c_i the original one and α the global watermarking strength which controls the energy of the embedded watermark. The main drawback is that the low-frequency coefficients are disturbed with the same amplitude than the higher frequency ones, which involve a larger visual distortion. Moreover, low-frequency coefficients are much higher and less sensitive to perturbations than high-frequency ones. At the opposite, the modulating scheme from Wu and Kobbelt is basically the following:

$$\hat{c}_i = c_i + c_i v_i \alpha. \tag{8}$$

Thus, the modulating amplitude is directly proportional to the coefficient value, therefore it will rapidly converge toward zero; indeed, the spectrum that they obtain with their decomposition is similar to ours (see Fig. 3). Thus, only very low-frequency coefficients will be considered in the watermarking process.

In order to avoid both drawbacks of these existing methods, we introduce a new coefficient modulation scheme: the *low-frequency favouring* (LFF) modulation, which favours low frequencies (of which alterations remain

nearly invisible for the human eyes), but also modulates higher frequency ones. Indeed, we have to exploit to the maximum the cover data since the subdivision control meshes have a quite small coefficient number. Our scheme is the following:

$$\hat{c}_i = c_i + v_i \alpha \beta_i, \tag{9}$$

with β_i, the local watermarking strength which adapts the modulation amplitude to the frequency:

$$\beta_i = \begin{cases} 1 & \text{if } i \geqslant T, \\ g*i + (1 - g*T) & \text{if } i < T. \end{cases} \tag{10}$$

T is a user-defined threshold (usually fixed to $n/10$, with n the number of coefficients), and g is the gradient of the linear approximation of the amplitude spectrum between coefficients 1 and T. The main idea is to have a constant watermark (α strength) for middle and high-frequency coefficients (index $> T$) and then increase linearly the strength (by increasing β) for low frequencies. Concerning the gradient g of the β function before T, we have calculated a linear approximation of the amplitude spectrum in $[1, T]$ and followed its gradient, in order to adapt the watermarking function to the considered object. Fig. 3 shows an example of β functions for the SubRabbit shape and for different T values.

Increasing the watermarking strength for low-frequency coefficients does not increase the visual distortion since the human eye is much more sensible to normal variations than to geometric modifications, like it was observed by Sorkine et al. [41]. Moreover, a high-frequency distortion applied on a subdivision control mesh implies a low-frequency distortion on the limit surface since a control mesh can be considered as a coarse low-frequency version of its associated limit surface. For instance, the 3D mesh wavelet theory [26] is based on subdivision inversion. This fact allows us to consider the whole spectra to embed the mark, contrary to existing algorithms which consider only very low-frequency coefficients [36], or the first half [34], in the embedding process.

3.3. Message sequence generation

Most of the existing algorithms ensure robustness to high-frequency attacks (noise addition, smoothing, simplification) by watermarking only very low frequencies [29,31,36]; however, these methods are not so robust to low-frequency attacks like non-uniform scaling or other global deformations. In a different way Ohbuchi et al. [33] repeat the mark along the spectra, and then average the extracted marks. Unfortunately, this technique is not an optimal way of adding redundancy and requires a lot of repetitions (≈ 10) to obtain a good robustness, which are not always possible for small meshes, such as subdivision control polyhedrons (see Fig. 1a).

3.3.1. Communication theory and ECC

A watermarking system can be viewed as a digital communication system [42]; indeed the 3D object represents the communication channel and the objective is to ensure the reliable transmission of the watermark message through this channel. Thus, like for a traditional communication system, it seems natural to consider the use of Error Correcting Codes (ECC) to increase the robustness of the transmission.

A lot of different ECC exist in the field of telecommunication: *repetition coding* (like Ohbuchi et al. do), *algebraic coding* (Hamming, BCH, etc.), *convolutional coding* (Viterbi, etc.) and *turbocodes*. Most of these existing ECC are characterized by their rate of redundancy rr which is the average number of bits necessary to encode 1 bit of the message. Thus $rr \times k$ bits are necessary to encode a message of length k.

These ECC also depend on the nature of the considered channel: *binary symmetric* or *additive white Gaussian noise*. A watermarking channel is said to be *binary symmetric* if the embedded binary message M is decoded to a binary code word, whereas it is considered to be *Gaussian* if one extracts a Gaussian real vector with mean M, that is our case.

Two decoding strategies exist for Gaussian channels: *hard decision decoding* or *soft decision decoding*. Hard decision decoding consists in thresholding the $rr \times k$ size extracted Gaussian real vector in a binary vector and then applying the error correction decoding. Of course, this decoding principle is not optimal since the thresholding implies a loss of valuable information. Better performance can be achieved by taking into account the real valued vector extracted directly from the Gaussian channel; that is precisely what soft decoding achieves.

Baudry et al. [43] have investigated the use of ECC, within the field of 2D image watermarking. Their conclusion highlights the contribution of such algorithms for the robustness and shows the significant superiority of convolutional codes associated with soft decision Viterbi decoding [44].

3.3.2. Convolutional encoding

During convolutional encoding, k input bits are mapped to m output bits to give a *rate* k/m coded bit stream. Each output bit is constructed not only from the current input bit but also using the $l - 1$ previous ones, by using l blocks of shift registers. Bits in registers are outputted to do binary modulo 2 additions, according to certain rules, whose results are the m output bits. l is called the *constraint length*. An example with $l = 3$, $k = 1$ and $m = 2$ is shown in Fig. 4.

A convolutional encoding can be expressed by means of a trellis diagram (see Fig. 5 for the trellis corresponding to the encoder from Fig. 4). Solid and broken lines show, respectively, code branches produced by $U_n = 0$ and 1 input bits. $U_{n-1}U_{n-2}$ expressions correspond to the different internal states of the encoder, while generated
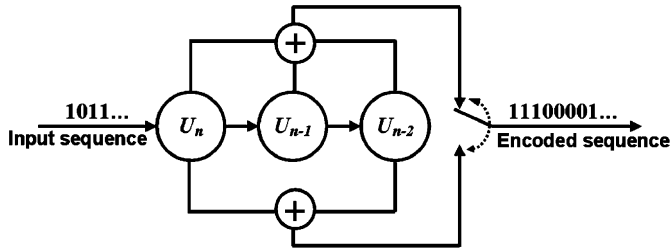
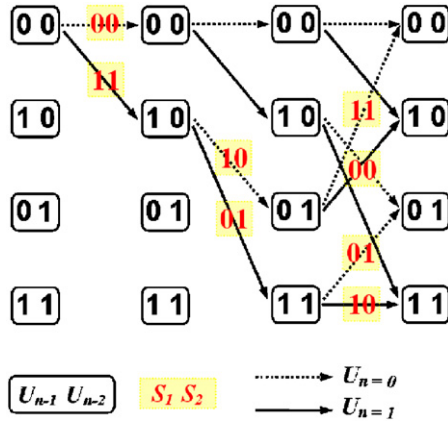Fig. 4. Convolutional encoder with $l = 3$, $k = 1$ and $m = 2$.

Fig. 5. Trellis diagram corresponding to convolutional encoder from Fig. 4.

Fig. 6. Viterbi decoding of the sequence 11 00 00 01. The result is 1011.

symbols are represented by the $S_1 S_2$ expressions. For example, the input sequence 1011 generates the encoded sequence 11 10 00 01; the encoding details are the following (the bits are read from left to right, U_n is underlined):

$$\begin{cases} 00\underline{1}011 \Rightarrow U_{n-1}U_{n-2} = 00 \Rightarrow S_1 S_2 = 11, \\ 001\underline{0}11 \Rightarrow U_{n-1}U_{n-2} = 10 \Rightarrow S_1 S_2 = 10, \\ 0010\underline{1}1 \Rightarrow U_{n-1}U_{n-2} = 01 \Rightarrow S_1 S_2 = 00, \\ 00101\underline{1} \Rightarrow U_{n-1}U_{n-2} = 10 \Rightarrow S_1 S_2 = 01. \end{cases} \quad (11)$$

3.3.3. Viterbi soft decoding

The Viterbi decoding algorithm [44] is a type of decoding algorithm used with convolutional encoding. This maximum likelihood decoder searches all the possible paths in the trellis and compares the metrics between each path and the input sequence. The path with the minimum metric is selected as the output.

Fig. 6 illustrates a Viterbi decoding of the code word 11 00 00 01, corresponding to the encoded input sequence 1011 (see previous section), with a transmission error at the third bit. Since our encoder has a rate $k/m = \frac{1}{2}$, the code word is read 2 bits by 2 bits. Starting from the initial 00 state, each possible path (corresponding to a possible decoded bit) is associated with an estimated transmitted 2 bits symbol which is compared with the really received symbol (at the bottom of Fig. 6). A distance is then calculated between these symbols. In our example the considered distance is the Hamming distance (D_H)
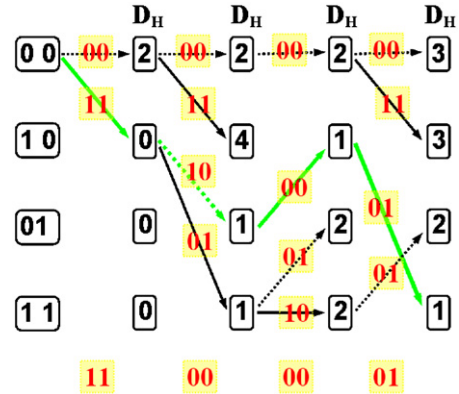
corresponding to the number of dissimilar bits. The path leading to the smallest distance ($D_H = 1$) gives us the decoded sequence: 1011 in our example.

In the case of an *additive white Gaussian noise channel* (our case), the code word to decode, after transmission is not a binary sequence but rather a real Gaussian vector, with mean as the original encoded sentence. Thus, distances associated to the possible paths during Viterbi decoding algorithm can be calculated using this real vector which carries much more information than a thresholded binary one: that corresponds to *soft decoding* and that is precisely what we use in our algorithm.

3.4. Control mesh synchronization

3.4.1. Context and presentation

The watermarked subdivision surface can be captured and/or attacked in a subdivided (i.e. smooth) version, thus we have to be able to retrieve the mark even in such a case. The subdivision mechanism is linear, thus, it seems easy to retrieve the corresponding control mesh by inverting the subdivision rules. Unfortunately, if the subdivided surface has been attacked (noise addition, remeshing), the inversion becomes impossible. Another solution is the subdivision-based wavelet decomposition [26], but it deals only with semi-regular triangular meshes.

Our solution comes from approximation theory: starting from the reference original subdivision surface, our objective is to move iteratively its control points in order to match it with the suspect smooth surface, we call this operation the *control mesh synchronization*.

This problem ties up with the subdivision surface approximation issue, which was investigated by several authors. Lee et al. [9] and Hoppe et al. [7] sample the input mesh with a set of points and minimize a quadratic error to the subdivision surface. Suzuki et al. [8] propose a faster approach: the position of each control point is optimized, only by reducing the distance between its limit position and the target surface. Hence, only subsets of the surfaces are involved in the fitting procedure, thus results are not so precise and may produce oscillations. Ma et al. [11] consider the minimization of the distance from vertices of the

subdivision surface after several refinements to the target mesh. Our algorithm follows this framework while using a point to surface distance minimization, based on the local quadratic approximant introduced by Pottmann and Leopoldseder [45], rather than a point to point distance minimization. This algorithm used for subdivision surface approximation by Lavoué et al. [12] and Marinov and Kobbelt [13] allows more accurate and rapid convergence.

The principal contribution of Pottmann and Leopoldseder [45] is the definition of local approximants of the squared distance from a point to a surface. Thus, the minimization of this point to surface distance is much faster than the traditional point to point distance. The local approximant of the point to surface quadratic distance is defined as follows: considering a smooth surface Ψ, we can define at each point t_0, a Cartesian system (e_1, e_2, e_3) whose first two vectors e_1, e_2 are the principal curvature directions and e_3 is the normal vector. Considering this frame, the local quadratic approximant $F_d(p)$ of the squared distance of a point p at $(0, 0, d)$ to the surface Ψ is given by [45]

$$F_d(x_1, x_2, x_3) = \frac{d}{d + \rho_1} x_1^2 + \frac{d}{d + \rho_2} x_2^2 + x_3^2, \tag{12}$$

where x_1, x_2 and x_3 are the coordinates of p with respect to the frame (e_1, e_2, e_3) and ρ_1 (resp. ρ_2) is the curvature radius at $\Psi(t_0)$, corresponding to the curvature direction e_1 (resp. e_2).

3.4.2. Algorithm

For a given target smooth surface (attacked subdivided watermarked surface) (see Fig. 7a) and a given reference subdivision control mesh (see Fig. 7b), this process aims at displacing control points by minimizing a global error between the corresponding limit surface and the target one. To achieve this purpose, we use a least square method
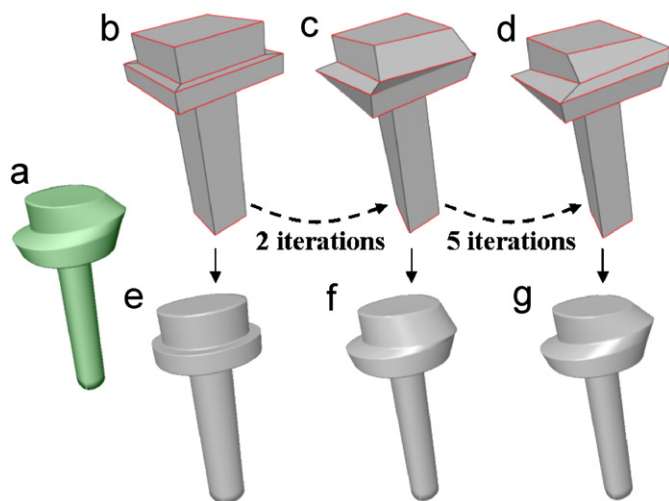

Fig. 7. Example of *synchronization*. (a) Suspect smooth surface, (b,c,d) reference control polyhedron after 0, 2 and 5 synchronization iterations, (e,f,g) corresponding limit surfaces.

based on the quadratic distance approximants defined by Pottmann and Leopoldseder [45] (see previous section). Our algorithm is the following:

- The curvature is calculated for each vertex of the target surface. We have implemented the work of Cohen-Steiner et al. [46], based on the normal cycle. This curvature estimation procedure has proven to be quite efficient and stable and gives very satisfying results even for bad tessellated objects.
- K sample points S_k are chosen on the reference subdivision surface, they correspond to vertices of the subdivided polyhedron at a finer level l_0. The associated footpoints (projections of the sample points on the target surface) are extracted. For each of them, we compute the curvature tensor, by a linear interpolation of those of the surrounding vertices, using barycentric coordinates. This tensor allows us to construct the frame e_1, e_2, e_3 and the curvature radii ρ_1 and ρ_2, useful for the point to surface distance computation (see Eq. (12)). The sample points S_k can be computed as linear combinations of the initial control points P_i^0 (see Section 2.1); they correspond to vertices $P_i^{l_0}$ at the finer level l_0:

$$S_k = C_k(P_1^0, P_2^0, \ldots, P_n^0). \tag{13}$$

- The functionals C_k are determined using iterative multiplications of the l_0 subdivision matrices associated with the subdivision rules.
- For all S_k, local quadratic approximants F_d^k of the squared distances to the target surface are expressed according to the frames e_1, e_2, e_3 at the corresponding footpoints. The minimization of their sum F gives the new positions of the control points P_i^0:

$$F = \sum_k F_d^k(S_k) = \sum_k F_d^k(C_k(P_1^0, P_2^0, \ldots, P_n^0)). \tag{14}$$

The minimization of this quadratic function leads to the resolution of a linear squared system.

Concerning the choice of the number K of sample points S_k, we choose $l_0 = 1$ ($K \approx n \times 4$) or $l_0 = 2$ ($K \approx n \times 16$), depending on the estimated subdivision level of the target subdivided surface. Indeed, this suspect surface may result from 1, 2 or even more subdivision steps. Since each subdivision step leads to sort of shrinkage of the surface, we have to estimate this number in order to conduct a correct approximation. Practically we distinguish three cases:

- No subdivision, thus no need of synchronization.
- One subdivision (thus we take $l_0 = 1$).
- Two or more subdivisions, then we consider $l_0 = 2$. Indeed, the shrinkage effect is almost invisible after the second subdivision iteration.

To detect the correct case, we consider the reference control polyhedron and subdivided versions (issued from 1 and 2

subdivision steps), and we take the version which minimizes the mean L1 error to the target suspect surface.

Fig. 7a presents a smooth surface coming from four subdivisions (and possibly attacks) of a watermarked control mesh (Catmull–Clark rules), thus we consider $l_0 = 2$. The watermark strength has been exaggerated for this experiment. The reference original subdivision surface is shown in Fig. 7b (control mesh) and 7e (limit surface). After only 5 synchronization iterations, the limit surface (Fig. 7g) is perfectly fitted with the suspect one (Fig. 7a). Resulting errors are respectively 3.84×10^{-3} and 0.03×10^{-3} after 2 and 5 iterations (surfaces were normalized in a cubic bounding box of length equal to 1). Thus, after 5 iterations we have retrieved the shape of the watermarked control mesh (see Fig. 7d), and we are able to launch the watermark extraction.

3.5. Complete insertion and extraction algorithms

3.5.1. Watermark insertion

Given a binary mark A of size k to embed, the first step is to produce an m bits code word $B = (b_1, b_2, \ldots, b_m)$ $(m > k)$, using convolutional encoding (see Section 3.3), in order to increase the mark robustness. The control mesh of the subdivision surface to watermark is then decomposed in the spectral domain (see Section 3.1) to produce the three spectral coefficient vectors P, Q and R of size n. The m dimensional watermark B $(m < n)$ will be embedded in the 3D subdivision control mesh, by modulating these spectral coefficients. In order to increase the robustness, we have chosen to repeat the watermark on each vector P, Q and R. Nevertheless, since a specific attack may alter the same part (high, low or middle frequencies) of each coordinate spectrum, we have chosen to shuffle the mark B into three different versions B^1, B^2, B^3 using three distinct random interleavers I_1, I_2, I_3:

$$\begin{cases} B^1 = I_1(B), \\ B^2 = I_2(B), \\ B^3 = I_3(B). \end{cases} \tag{15}$$

Before embedding the binary sequences B^1, B^2 and B^3, we first construct the three modulating vectors $B^{1'}$, $B^{2'}$ and $B^{3'}$ by the following mapping:

$$b_i^{x'} = \begin{cases} -1 & \text{if } b_i^x = 0, \\ 1 & \text{if } b_i^x = 1, \end{cases} \quad i = 1 \ldots m, \quad x = 1, 2, 3. \tag{16}$$

These vectors are then inserted, respectively, in the spectral components P, Q and R using our coefficient modulation algorithm (see Section 3.2):

$$\begin{aligned} \hat{p}_i &= p_i + b_i^{1'} \alpha \beta_i, \\ \hat{q}_i &= q_i + b_i^{2'} \alpha \beta_i, \quad i = 1 \ldots m. \\ \hat{r}_i &= r_i + b_i^{3'} \alpha \beta_i, \end{aligned} \tag{17}$$

3.5.2. Watermark extraction

The extraction process needs the watermarked (and possibly attacked) control mesh, the original reference control mesh (i.e. the algorithm is non-blind) and the α and T values.

The watermarked control mesh and the original one are first aligned by a registration process [47]. If the watermarked subdivision surface is in a subdivided form, we first align it with the corresponding subdivided version of the original surface (issued from one or two subdivisions, see Section 3.4), and then apply the synchronization process to retrieve the corresponding watermarked control mesh.

Both watermarked and reference control meshes are then decomposed on the eigenvector basis computed on the reference one. Thus, we obtain, respectively, spectral coefficients vectors $\hat{P}$, $\hat{Q}$ and $\hat{R}$ and P, Q and R.

Firstly, we extract three real number vectors $\hat{B}^1$, $\hat{B}^2$ and $\hat{B}^3$ computed as follows:

$$\begin{aligned} \hat{b}_i^1 &= \frac{\hat{p}_i - p_i}{\alpha \beta_i}, \\ \hat{b}_i^2 &= \frac{\hat{q}_i - q_i}{\alpha \beta_i}, \quad i = 1 \ldots m. \\ \hat{b}_i^3 &= \frac{\hat{r}_i - r_i}{\alpha \beta_i}. \end{aligned} \tag{18}$$

Then, these vectors are desinterleaved and summed to give the m dimensional real number vector $\hat{B}$:

$$\hat{B} = \tfrac{1}{3}(I_1^{-1}(\hat{B}^1) + I_2^{-1}(\hat{B}^2) + I_3^{-1}(\hat{B}^3)). \tag{19}$$

This vector is then decoded using the soft Viterbi algorithm to give the extracted mark $\hat{A} = (\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_k)$ of size k. The correlation, between the embedded watermark A and the extracted one, is defined by

$$correlation = \sum_{i=1}^{i=k}(C_i), \quad C_i = \begin{cases} 1 & \text{if } \hat{a}_i = a_i, \\ -1 & \text{if } \hat{a}_i \neq a_i. \end{cases} \tag{20}$$

4. Experiments and results

We have conducted experiments on several different subdivision surfaces. Examples are given for three typical objects, from different natures and coming from different creation processes. They are illustrated in Fig. 8 (control meshes at the top and limit surfaces at the bottom):

(a) *SubPlane* (154 control points), whose control mesh was directly hand designed. It is composed of a majority of quadrangles.
(b) *SubRabbit* (200 control points), whose control mesh comes from the subdivision surface approximation algorithm from Kanai [10] applied on the well-known natural object Stanford Bunny. It is a triangle only control mesh.
(c) *SubFandisk* (86 control points), whose control mesh comes from the subdivision surface approximation algorithm from Lavoué et al. [12] applied on the
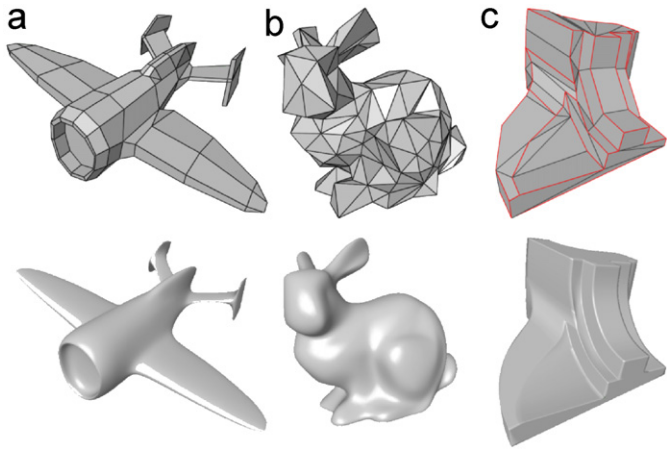
Fig. 8. Subdivision surfaces used in our experiments. Control meshes at the top and limit surfaces at the bottom. (a) Sub Plane, (b) Sub Rabbit, (c) Sub Fandisk.

Table 1
Computation times of the different steps of our watermarking algorithm

	SubFandisk	SubPlane	SubRabbit
Vertex/face number	86/101	154/161	200/384
Spectral decomposition (s)	0.109	0.328	0.766
Watermark insertion (s)	0.093	0.125	0.156
Watermark extraction (s)	0.031	0.032	0.047

well-known mechanical Fandisk object. It is composed of triangles, quadrangles and higher order polygons.

These subdivision surfaces are associated with subdivision rules from Stam and Loop [6]; this hybrid quad/triangle scheme reproduces Catmull–Clark on quad regions and Loop on triangle regions.

In all our experiments, we have considered the embedding of a watermark of length $k = 32$ bits and with parameters $T = 10$ and $\alpha = 0.005$. Every object is scaled to a bounding box of length equal to 1.

The computational cost of the algorithm is first discussed. Then we will study the visual distortion associated with our watermarking scheme. Finally, the robustness will be verified for diverse attacks directed against both the control mesh and the limit surface.

4.1. Timing analysis

Table 1 presents computation times for the watermarking scheme applied on the three subdivision surfaces. The mark length is $m = 32$ bits and the rate of the convolutional coder is $\frac{1}{3}$ (96 coefficients are watermarked on each coordinate spectrum) for SubPlane and SubRabbit and $\frac{1}{2}$ (64 coefficients are watermarked) for SubFandisk. All experiments were conducted on a PC with a 2 GHz XEON bi-processor.

Since our control meshes have quite small numbers of vertices (much smaller than dense mesh versions of the same shapes), the spectral decomposition is fast (less than 1 s). Moreover, the watermark insertion and extraction mechanisms, based on spectral coefficient modulation, are very simple and thus also quite fast.

4.2. Error distortion

The visual distortion introduced by the watermark embedding is quite critical in a watermarking algorithm;

indeed the mark has to be nearly invisible for a human eye, considering the subdivision control mesh and above all the limit surface. That is why most of the existing spectral schemes only modulate low-frequency coefficients. In our case, since we watermark the control polyhedron of the subdivision surface, we have asserted that modulating middle and high frequencies of this coarse mesh will only have a low-frequency effect on the limit surface and thus will remain nearly invisible for a human eye.

Fig. 9 shows the SubFandisk subdivision surface, and the watermarked versions associated, respectively, with the simple modulating scheme used by Ohbuchi et al. [33,34] (which applies the same strength on every spectral coefficients) and our LFF scheme which increases the strength on low frequencies (see Fig. 3). For both versions, the value of α is 0.005, and the rate is $\frac{1}{2}$, thus we have altered 64 coefficients.

The distance maps associated with both watermarked versions are illustrated in Fig. 9c; they represent, for each vertex of the watermarked limit surfaces, its Hausdorff distance to the original limit surface. Table 2 details global geometric error values between watermarked limit surfaces and the original one. It is admitted in the psychovisual research community that standard geometric distances do not match well with the human visual perception. Accordingly, Table 2 also presents results of the perceptual distortion measure MSDM from Lavoué et al. [48], which provides an approximation of the subjective visual similarity. This value tends toward 1 (theoretical limit) when the objects are visually different and is equal to 0 for identical ones.

The first remark is that the watermark is nearly invisible both on watermarked control meshes and limit surfaces. Particularly, the LFF modulation scheme (on the right) is not more visible than the simple one, whereas it provides a better robustness by more strongly modulating the low frequencies. Indeed, geometric errors are of course larger for the LFF modulation (since the modulation strength is larger on low coefficients), but the perceptual distance to the original surface (MSDM value) is the same for both.

The second remark is that, even when high frequencies of the object spectrum (64 coefficients, among 86) are altered, the corresponding distortions on the limit surfaces have a quite low-frequency aspect; in particular, the limit surface remains smooth, what is critical particularly in the case of CAD models where even slight distortions can be visually
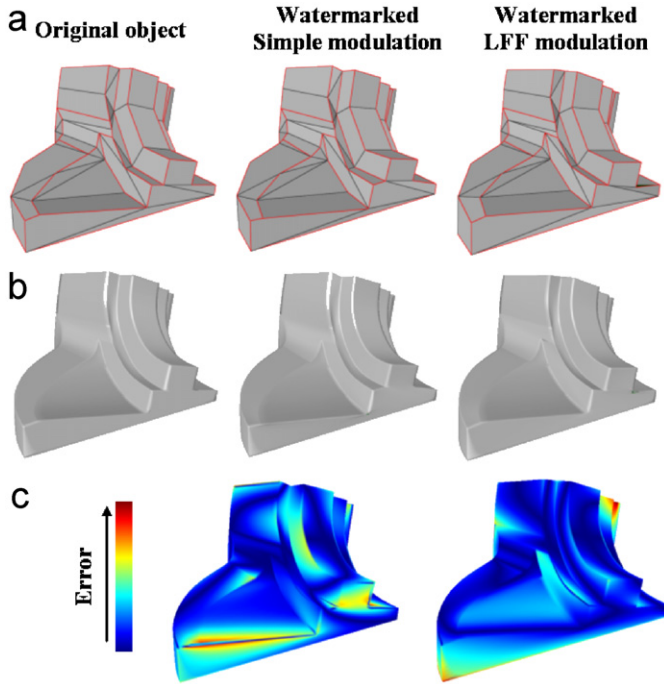
Fig. 9. *SubFandisk* subdivision surface. Original, watermarked with simple modulation, watermarked with LFF modulation. (a) Control meshes, (b) limit surfaces, (c) distance maps with the original limit surface.

Table 2
Errors between watermarked limit surfaces and the original one (for the *SubFandisk* object): mean (L_1), root mean square (L_2), maximum (L_∞) and the perceptual measure (MSDM) from Lavoué et al. [48]

	L_1 (10^{-3})	L_2 (10^{-3})	L_∞ (10^{-3})	MSDM
Simple modulation	1.98	2.58	10.98	0.28
LFF modulation	2.49	3.25	13.69	0.28

damaging. These first experiments have validated our assertions that:

- Our LFF modulation scheme is not more visible than the simple additive one.
- We can watermark all frequencies of the control meshes (not just low frequencies like most of the mesh watermarking schemes), since this will not visually much alter the limit surface.

Benedens et al. [49] present a study of different criteria to respect, in order to ensure imperceptibility of a watermark. They emphasize the importance of preserving the continuity of the surface, what is achieved by our algorithm, but also the symmetry of the object. Since we consider a spectral decomposition which does not integrate geometric information, our scheme unfortunately does not preserve symmetry. This issue could be solved by a previous detection of symmetric features and a decomposition of the object.

4.3. Attacks against the control mesh

Our watermarking scheme can be considered as an improvement of the mesh watermarking algorithm from Ohbuchi et al. [33], by firstly introducing a new modulation algorithm (the LFF scheme) and secondly by modulating the binary message by convolutional encoding. Thus, we have established the efficiency of these improvements by checking robustness against three types of real-world attacks which alter different parts of the object spectrum: noise addition (rather high frequencies), non-uniform scaling (rather low frequencies) and quantization (rather high frequencies). These attacks are illustrated in Fig. 10. For each attack, we consider four algorithms:

- Simple modulation, repetition coding (basically the Ohbuchi scheme).
- Simple modulation, convolutional coding.
- LFF modulation, repetition coding.
- LFF modulation, convolutional coding (basically our complete scheme).

In the following experiments, for each correlation value presented in Figs. 11–13, we have repeated 100 times the insertion, the attack and the extraction with random bit patterns of length 32 bits and then averaged the obtained correlations.

We have not considered connectivity alterations, like simplification or remeshing, since changing the connectivity of a control polyhedron would completely change the limit surface and thus do not represent a realistic real-world attack.
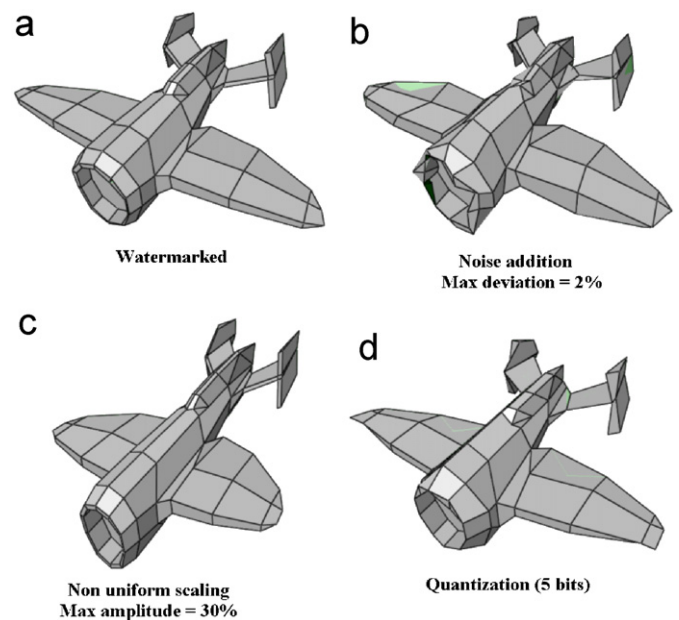

Fig. 10. Watermarked SubPlane model, and various attacks for which the extracted correlation is 100%.
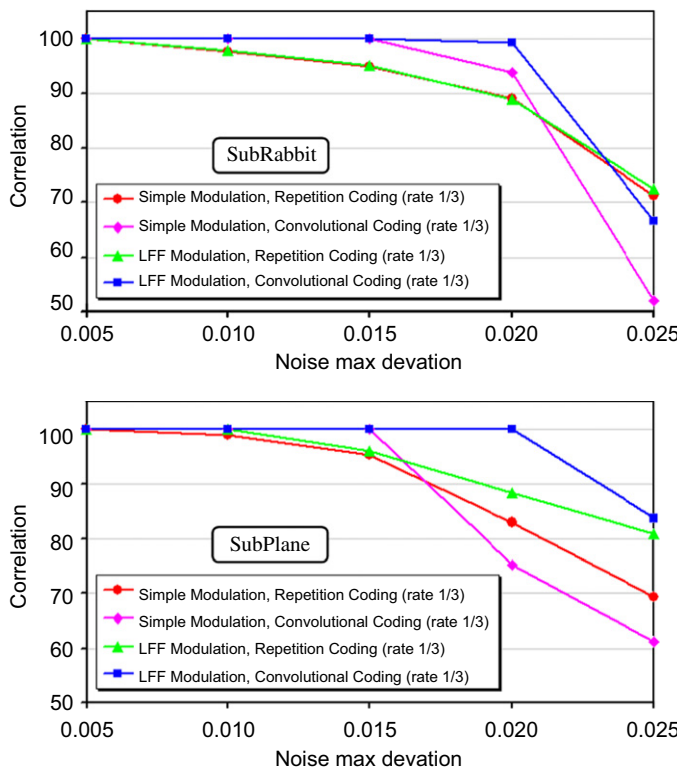
Fig. 11. Watermarking correlation (%) of the *SubRabbit* and *SubPlane* objects under noise addition attacks with increasing maximum deviations.
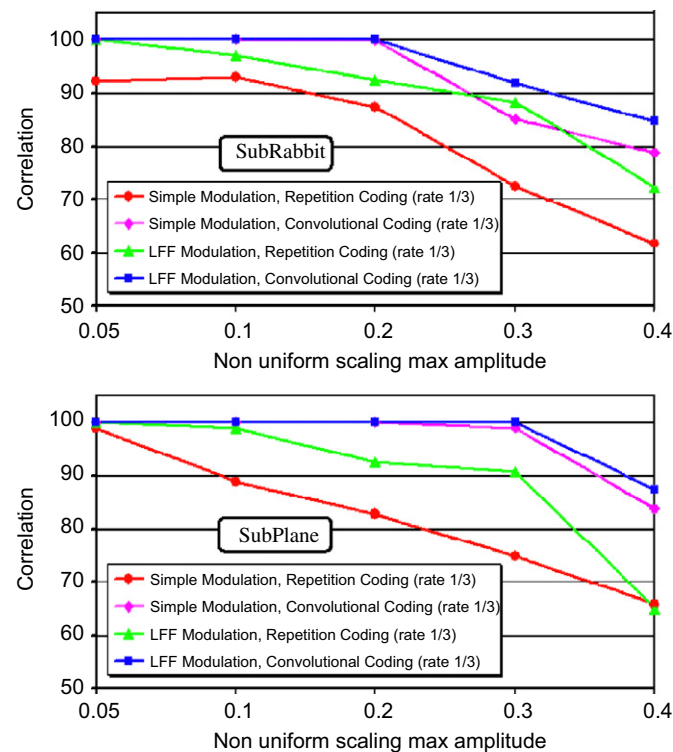

Fig. 12. Watermarking correlation (%) of the *SubRabbit* and *SubPlane* objects under non-uniform scaling attacks with increasing maximum amplitudes.

4.3.1. Noise addition

We modify the three coordinates of each vertex of the control mesh, according to a randomly chosen offset between 0 and a maximum deviation E_{max}. Fig. 11 shows the extracted average correlation, according to increasing E_{max} values, for the SubPlane and the SubRabbit objects. Notice that, for both objects, the LFF modulation and the watermark convolutional encoding significantly improve robustness. For the SubRabbit and the SubPlane, the correlation reaches 100% for $E_{max} = 0.020$ (four times the value of the watermark strength α), while the basic scheme (simple modulation, repetition coding) gives 90% and 85%, respectively.

4.3.2. Non-uniform scaling

For each axes (X, Y and Z), we compute a scaling value, randomly chosen between $1 - S_{max}$ and $1 + S_{max}$, and we multiply the corresponding coordinates by this value. Fig. 12 shows the extracted average correlation, according to increasing S_{max} values, for the SubPlane and the SubRabbit objects. Like for the noise addition attack, our complete scheme gives much better results than the simple one. For $S_{max} = 0.3$ (coordinates are multiplied by numbers between 1.3 and 0.7) we obtain for SubRabbit: 92% (our scheme) against 72% (the basic scheme), and for SubPlane: 100% against 75%.

4.3.3. Quantization

Quantization is a common step in most of the existing 3D compression techniques, thus we have to ensure robustness against such an attack. Fig. 13 shows the extracted average correlation, for different quantizations associated with decreasing numbers of bits. For the subFandisk and the SubPlane, our scheme gives 100% average correlations for a 5 bits quantization, whereas the simple scheme from Ohbuchi et al. gives, respectively, 79% and 92%.

4.4. Attacks against the limit surface

Since a suspect subdivision surface can be retrieved in a subdivided form, we have tested the robustness of the watermarking scheme to the synchronization process and to some attacks against a subdivided watermarked surface.

4.4.1. Connectivity alterations

For this experiment, we have inserted a 32 bits mark into the SubFandisk object ($\alpha = 0.005$ and $rate = \frac{1}{2}$), and then applied 2 subdivision iterations. We have then considered three attacks: a triangulation of the faces (see Fig. 14a), and 2 rather strong simplification steps (see Figs. 14b and c).

For these three versions, after synchronization ($l_0 = 2$) and mark extraction, the retrieved correlation is 100%. Even if the synchronization step does not perfectly fit the target surface, our scheme is robust to the induced approximation error. Moreover, this experiment has
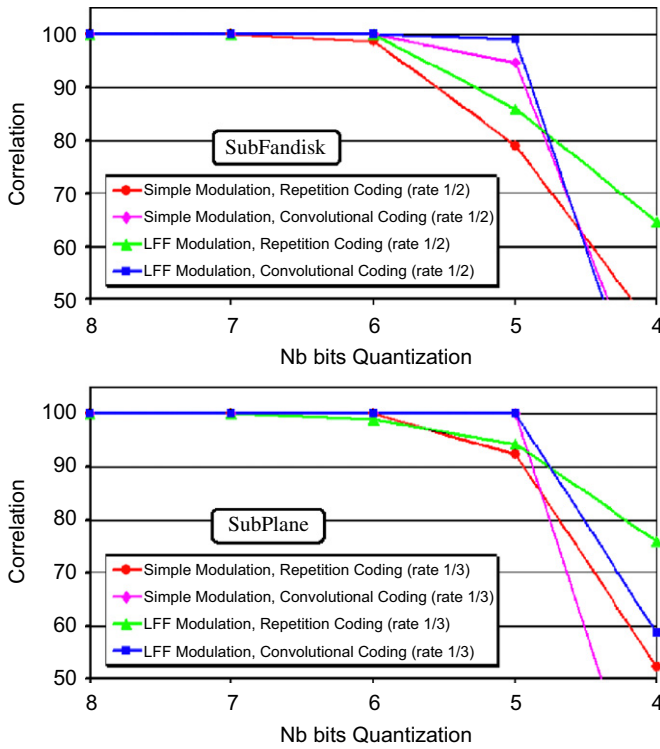
Fig. 13. Watermarking correlation (%) of the *SubFandisk* and *SubPlane* objects submitted to different quantizations.

Fig. 15. Watermarked *SubFandisk* object after 3 subdivision iterations (a) and noise addition (b). The extracted correlation is 100%.
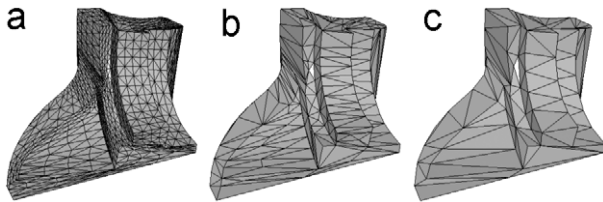

Fig. 14. Watermarked *SubFandisk* object after 2 subdivision iterations and triangulation (a), and 2 simplification steps (b,c). The extracted correlation is 100%. (a) 1282 vertices, (b) 270 vertices, (c) 110 vertices.

coefficients of the subdivision control mesh. Due to the compactness of the cover object (a coarse control mesh), our algorithm optimizes the trade-off between watermarking redundancy and imperceptibility by modulating coefficients according to a new scheme (LFF) and by using ECC. Experiments have shown an average 20% improvement of the robustness, compared with a standard modulation scheme [34].

Since a watermarked subdivision surface can be captured and/or attacked in a subdivided (i.e. smooth) version, we have also introduced a synchronization process allowing to retrieve the corresponding control mesh and to correctly extract the mark. This process provides efficient robustness against remeshing or simplification attacks which are considered critical issues according to evaluation criteria from Benedens et al. [49].

Concerning future work, we plan to investigate other types of ECC, providing robustness even for very severe attacks, since convolutional coding efficiency tends to fall down in such cases. It should be useful to modelize the spectral distortion introduced by the different types of attacks (noise addition, quantization, scaling, etc.) in order to construct specific correcting schemes.

Finally, the properties of subdivision surfaces could be exploited to design blind watermarking schemes; an interesting idea could be to use connectivity properties (semi-regularity) of limit surfaces.

proven the high robustness to hard connectivity alterations like the simplifications illustrated in Figs. 14b and c.

4.4.2. Geometry alterations

In order to establish the robustness of our scheme to geometric attacks against the subdivided watermarked surface, we have considered the watermarked SubFandisk object ($\alpha = 0.005$ and $rate = \frac{1}{2}$), after three subdivision iterations (see Fig. 15a) and noise addition (see Fig. 15b). The synchronization is processed with $l_0 = 2$, and we obtain for both case a 100% correlation, after the mark extraction. Even if the noise amplitude is not very high (max. deviation $= (0.4\%)$), this result is quite satisfying since we have just watermarked the coarse control polyhedron (86 vertices).

5. Conclusion

We have presented a robust watermarking scheme for subdivision surfaces, based on the modulation of spectral
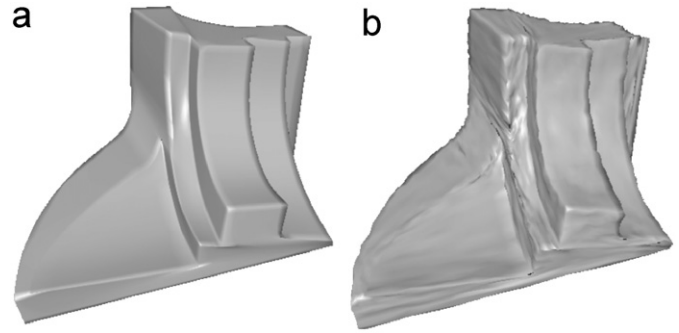
References

[1] Cox I-J, Miller M-L, Bloom J-A. Digital watermarking. Los Altos, CA: Morgan Kaufman; 2002.

[2] MPEG4, Iso-iec 14496-16. Coding of audio-visual objects: animation framework extension (afx), 2006.

[3] Doo D, Sabin M. Behavior of recursive division surfaces near extraordinary points. Computer Aided Design 1978;10:356–60.

[4] Catmull E, Clark J. Recursively generated b-spline surfaces on arbitrary topological meshes. Computer-Aided Design 1978;10(6): 350–5.

[5] Loop C. Smooth subdivision surfaces based on triangles. Master's thesis, Utah University, 1987.

[6] Stam J, Loop C. Quad/triangle subdivision. Computer Graphics Forum 2003;22(1):79–85.

[7] Hoppe H, DeRose T, Duchamp T, Halstead M, Jin H, McDonald J, et al. Piecewise smooth surface reconstruction. In: ACM Siggraph, vol. 28; 1994. p. 295–302.

[8] Suzuki H. Subdivision surface fitting to a range of points. In: IEEE Pacific graphics; 1999. p. 158–67.

[9] Lee A, Moreton H, Hoppe H. Displaced subdivision surfaces. In: ACM Siggraph; 2000. p. 85–94.

[10] Kanai T. Meshtoss—converting subdivision surfaces from dense meshes. In: The sixth international workshop on vision, modeling and visualization; 2001. p. 325–32.

[11] Ma W, Ma X, Tso S, Pan Z. A direct approach for subdivision surface fitting from a dense triangle mesh. Computer Aided Design 2004;36(16):525–36.

[12] Lavoué G, Dupont F, Baskurt A. A framework for quad/triangle subdivision surface fitting: application to mechanical objects. Computer Graphics Forum 2007;26(1) to appear.

[13] Marinov M, Kobbelt L. Optimization methods for scattered data approximation with subdivision surfaces. Graphical Models 2005; 67(5):452–73.

[14] Ohbuchi R, Masuda H, Aono M. A shape-preserving data embedding algorithm for nurbs curves and surfaces. In: IEEE computer graphics international; 1999. p. 180–7.

[15] Lee J-J, Cho N-I, Lee S-U. Watermarking algorithms for 3d nurbs graphic data. EURASIP Journal on Applied Signal Processing 2004;14:2142–52.

[16] Ohbuchi R, Masuda H, Aono M. Watermarking three-dimensional polygonal meshes. In: ACM multimedia; 1997. p. 261–72.

[17] Ohbuchi R, Masuda H, Aono M. Watermarking three-dimensional polygonal models through geometric and topological modifications. IEEE Journal on Selected Areas in Communication 1998;16(4): 551–9.

[18] Yeo B-L, Yeung M-M. Watermarking 3d objects for verification. IEEE Computer Graphics and Applications 1999;19(1):36–45.

[19] Cayre F, Macq B. Data hiding on 3-d triangle meshes. IEEE Transactions on Signal Processing 2003;51(4):939–49.

[20] Benedens O. Geometry-based watermarking of 3d models. IEEE Computer Graphics and Application 1999;19(1):46–55.

[21] Benedens O, Busch C. Toward blind detection of robust watermarks in polygonal models. Computer Graphics Forum 2000;19: 199–208.

[22] Yu Z, Ip H-H, Wok L-F. A robust watermarking scheme for 3d triangular mesh models. Pattern Recognition 2003; 36(11):2603–14.

[23] Maret Y, Ebrahimi T. Data hiding on 3d polygonal meshes. In: ACM workshop on multimedia and security; 2004. p. 68–74.

[24] Zafeiriou S, Tefas A, Pitas I. Blind robust watermarking schemes for copyright protection of 3d mesh objects. IEEE Transactions on Visualization and Computer Graphics 2005;11(5):596–607.

[25] Kanai S, Date H, Kishinami T. Digital watermarking for 3d polygons using multi-resolution wavelet decomposition. In: IFIP WG 5.2 International workshop on geometric modeling: fundamental and application (GEO-6); 1998. p. 296–307.

[26] Lounsbery M, DeRose T-D, Warren J. Multiresolution analysis for surfaces of arbitrary topological type. ACM transactions on graphics 1997;16(1):34–73.

[27] Uccheddu F, Corsini M, Barni M. Wavelet-based blind watermarking of 3d models. In: ACM workshop on multimedia and security; 2004. p. 143–54.

[28] Kim M-S, Valette S, Jung H-Y, Prost R. Watermarking of 3d irregular meshes based on wavelet multiresolution analysis, Lecture notes in computer Science, vol. 3710. In: Proceedings of IWDW, 2005. p. 313–24.

[29] Praun E, Hoppe H, Finkelstein H. Robust mesh watermarking. In: Siggraph; 1999. p. 69–76.

[30] Hoppe H. Progressive meshes. In: ACM Siggraph; 1996. p. 99–108.

[31] Yin K, Pan Z, Shi J, Hang D. Robust mesh watermarking based on multiresolution processing. Computers and Graphics 2001;25(3): 409–20.

[32] Guskov I, Sweldens W, Shroder P. Multiresolution signal processing for meshes. In: ACM Siggraph; 1999. p. 49–56.

[33] Ohbuchi R, Takahashi S, Miyazawa T, Mukaiyama A. Watermarking 3d polygonal meshes in the mesh spectral domain. In: Graphic interface; 2001. p. 9–17.

[34] Ohbuchi R, Mukaiyama A, Takahashi S. A frequency-domain approach to watermarking 3d shapes. Computer Graphic forum 2002;21(3):373–82.

[35] Karni Z, Gotsman C. Spectral compression of mesh geometry. In: ACM Siggraph; 2000. p. 279–86.

[36] Wu J, Kobbelt L. Efficient spectral watermarking of large meshes with orthogonal basis functions. The Visual Computers 2005; 21(8–10):848–57.

[37] Li L, Zhang D, Pan Z, Shi J, Zhou K, Kai Y. Watermarking 3d mesh by spherical parameterization. Computers and Graphics 2004; 28(6):981–9.

[38] Cayre F, Rondao-Alface P, Schmitt F, Macq B, Maître H. Application of spectral decomposition to compression and watermarking of 3d triangle mesh geometry. Signal Processing : Image Communications 2003;18(4):309–19.

[39] Bollabás B. Modern graph theory. Berlin: Springer; 1998.

[40] Cox I-J, Killian J, Leighton T, Shamoon T. Secure spread spectrum watermarking for multimedia. IEEE Transactions on Image Processing 1997;12(6):1673–87.

[41] Sorkine O, Cohen-Or D, Toldeo S. High-pass quantization for mesh encoding. In: Eurographics symposium on geometry processing; 2003. p. 42–51.

[42] Cox I-J, Miller M-L, McKellips A-L. Watermarking as communications with side information. Proceedings of the IEEE 1999;87(7): 1127–41.

[43] Baudry S, Delaigle J-F, Sankur B, Macq B, Maitre H. Analyses of error correction strategies for typical communication channels in watermarking. Signal Processing 2001;81(6):1239–50.

[44] Viterbi A. Error bounds for convolution codes and an asymptotically optimum decoding algorithm. IEEE Transactions on Information Theory 1967;13:260–9.

[45] Pottmann H, Leopoldseder S. A concept for parametric surface fitting which avoids the parametrization problem. Computer Aided Geometric Design 2003;20(6):343–62.

[46] Cohen-Steiner D, Morvan J. Restricted delaunay triangulations and normal cycle. In: 19th annual ACM symposium computation geometry, 2003.

[47] Besl P-J, McKay N-D. A method for registration of 3-d shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 1992;14(2):239–56.

[48] Lavoué G, Drelie Gelasca E, Dupont F, Baskurt A, Ebrahimi T. Perceptually driven 3d distance metrics with application to watermarking. In: SPIE applications of digital image processing XXIX, vol. 6312; 2006.

[49] Benedens O, Dittmann J, Petitcolas F. 3d watermarking design evaluation. In: SPIE security and watermarking of multimedia contents V, vol. 5020; 2003. p. 337–48.