Digital Watermarking of Compressed 3D Meshes

Florence Denis, Guillaume Lavoué, Florent Dupont, Atilla Baskurt

LIRIS UMR 5205 CNRS, Université Claude Bernard Lyon 1, Villeurbanne, France
{Florence.Denis, Guillaume.Lavoue, Florent.Dupont, Atilla.Baskurt}@liris.cnrs.fr

***Abstract*— This paper presents a robust watermarking algorithm applied to 3D compressed polygonal meshes. Copyright protection of 3D models becomes very important for many applications using public networks. As some recent compression techniques allow very high compression rates, it is of interest to verify that watermarking techniques support this kind of attack. In this paper we present the complete scheme developed: the compression algorithm, the watermarking algorithm and the mark extraction process.**

I. Introduction

At the present time three dimensional models are more and more present on private or public networks. In this context, the need for efficient tools to protect the copyright becomes even more acute. In recent times, the watermarking problematic has essentially focused on audio, images and videos. 3D model watermarking is an emergent and very challenging technique.

The context of our work is the SEMANTIC-3D project (http://www.semantic-3d.net) whose principal issue is the transmission of 3D mechanical models through low bandwidth channels in a visualization objective on various terminals. The 3D model database to handle comes from the car manufacturer Renault, and contains thousands of polygonal meshes representing CAD parts. We used a framework, based on subdivision surface approximation for efficient compression of 3D models represented by polygonal meshes. The algorithm fits a piecewise smooth subdivision surface to the input 3D mesh. The found control polyhedron is much more compact than the original mesh and visually represents the same shape after several subdivision steps. This control polyhedron is then encoded specifically to give the final compressed stream. This method is particularly suited for compression of meshes issued from mechanical or CAD parts.

As in 2D with the JPEG framework, it is interesting to develop new watermarking techniques in the compressed space. Hence, the robustness associated to the framework could be improved.

In section 2, a state of the art of 3D objects watermarking and compression is presented. Section 3 details the 3D compression method and section 4 focuses on the watermarking algorithm. Experiments have been conducted on CAD models and results are presented in section 5. A conclusion and some perspectives are given in the section 6.

II. State of the Art

A. *Compression*

A lot of work has been done about 3D object compression and particularly about polygonal meshes compression. This representation contains two kinds of information: geometry and connectivity, the first describing coordinates of the vertices in the 3D space, and the later describing how to connect these positions. The connectivity is often encoded using a region growing approach based on faces, edges [1] or vertices [2]. Fewer efforts have been done about geometry compression which is often simply performed by predictive coding and quantization. Other researches have put more efforts on geometry driven mesh coding, using wavelets [3] or spectral compression [4]. On the whole, better mesh compression methods give between 1 and 2 bytes per vertex; although this represents an excellent result, the output bit stream remains large for complex objects because of the high number of vertices to encode. Moreover lossy compression schemes [3] [4] often produce artifacts, visually damaging for smooth mechanical objects. That is why we have developed a new algorithm, based on subdivision surface fitting for efficiently compressing 3D meshes, for low bandwidth transmission and storage.

B. *3D objects watermarking*

Digital watermarking hides an embedded information in data in order to protect copyright or data integrity. The invisibility of the hidden message is the main constraint. For integrity assessment, a fragile watermarking is necessary: the embedded watermark disappears partially if the data is modified (attacked). At the opposite, for copyright protection applications, the main desired property for the watermarking technique is robustness. The capacity (number of bits that can be embedded) is also of interest to efficiently protect the mark from attacks.

1D and 2D watermarking schemes are generally distinguished by the working domain used for the message insertion: some of them directly modify the signal in the time (1D) or space (2D) domain, while others transform the cover data (signal or image to be watermarked) in a dual space representation (Fourier, wavelet, …) before inserting the mark.

Existent 3D mesh watermarking schemes are also based on different insertion domains, but also exploit either the geometrical characteristics of the 3D model (vertices coordinates), or its topological properties (connectivity).

Ohbuchi et al. [5] proposed several watermarking methods for 3D meshes, either based on coordinates displacements modifying triangles or tetrahedra characteristics or inserting new triangles into the mesh in a visible way. Kanai et al. [6] introduced multiresolution wavelet decomposition to embed the watermark in the large wavelet coefficients at various resolution levels, while Yin et al. [7] adopted a Guskov's multiresolution signal processing method with progressive local smoothing. Praun et al. [8] used a multiresolution analysis approach and a spread spectrum method to insert a message in the mesh spatial coordinates. In 2001, Ohbuchi et al. [9] proposed another watermarking method in the mesh spectral domain. More recently, Li et al. [10] used a spherical parameterization method where the mesh is iteratively simplified into a convex mesh before a spherical harmonic transformation is applied. The watermark is embedded into the spherical harmonic coefficients.

Robust techniques are those based on geometry, since topological modifications are destroyed by elementary simplification or re-meshing operations.

III. 3D COMPRESSION METHOD BASED ON SUBDIVISION SURFACES

A. *Subdivision surface presentation*

The basic idea of subdivision is to define a smooth shape from a coarse polyhedron by repeatedly and infinitely adding new vertices and edges according to certain subdivision rules. Many subdivision rules exist, some of them are adapted for triangular control meshes, like Loop [11] and others are adapted for quadrilateral ones, like Catmull-Clark [12]. Moreover special rules have been introduced to handle *sharp* edges. Subdivision surfaces offer many benefits: Firstly, they can be generated from arbitrary meshes (arbitrary topology), this implies no need of trimming curves (which are necessary for NURBS). Secondly, they can be generated at any level of detail, according to the terminal capacity for instance. And thirdly, subdivision surfaces are at least C1 continuous (except around *sharp* edges of course). Within our approximation framework, we have chosen the hybrid quad/triangle scheme developed by Stam and Loop [13] and illustrated on figure 1.

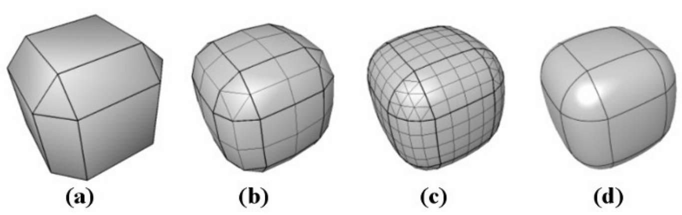


Figure 1. Example of quad/triangle subdivision. (a) Control mesh, (b,c) One and two subdivision steps, (d) Limit surface.

B. *Overview of our compression method*

Our framework for compression of 3D models is detailed in [14] and illustrated on figure 3: Firstly the target 3D object is segmented into surface patches according to the method described in [15]. The segmented patches have a near constant curvature and smooth boundaries (they are illustrated in green on figure 2.a). Then, the network of boundaries is extracted (see figure 2.b) and approximated with piecewise smooth subdivision curves (this approximation algorithm is described in [16]). A subdivision curve is a smooth curve represented by a coarse control polygon, thus, the boundary network approximation produces a network of control polygons (see figure 2.c) (*sharp* edges are in red). Next, for each patch an approximating subdivision surface is created by linking its boundary control points (extracted from the network) with respect to the lines of curvature of the target patch. The control mesh defining the whole surface is then created assembling every local control meshes (see figure 2.d). After several refinement steps, the subdivision surface will visually represent the shape of the original mesh (see figure 2.e). This control mesh is then encoded specifically to give the output binary stream. Hence the 3D model, once approximated, will be transmitted in the form of an encoded coarse polyhedron and, at the reception, displayed to any resolution, according to the terminal capacity, by iterative subdivisions. Note that this decompression process is very simple and therefore adapted for mobile terminals. Figure 3 shows this application scheme.

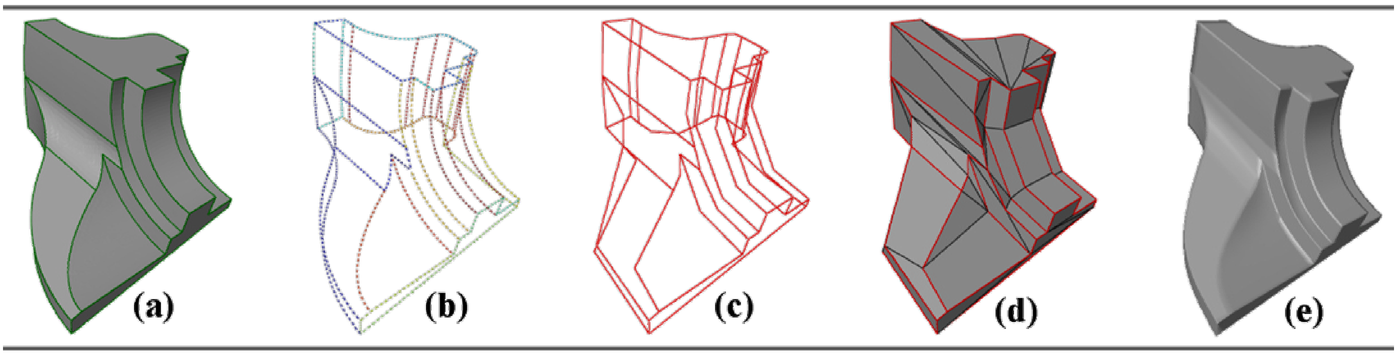


Figure 2. The different steps of our fitting scheme for the Fandisk object. (a) Segmentation, (b) Boundaries extraction, (c) Boundaries approximation, (d) Subdivision control mesh, (e) Limit surface.

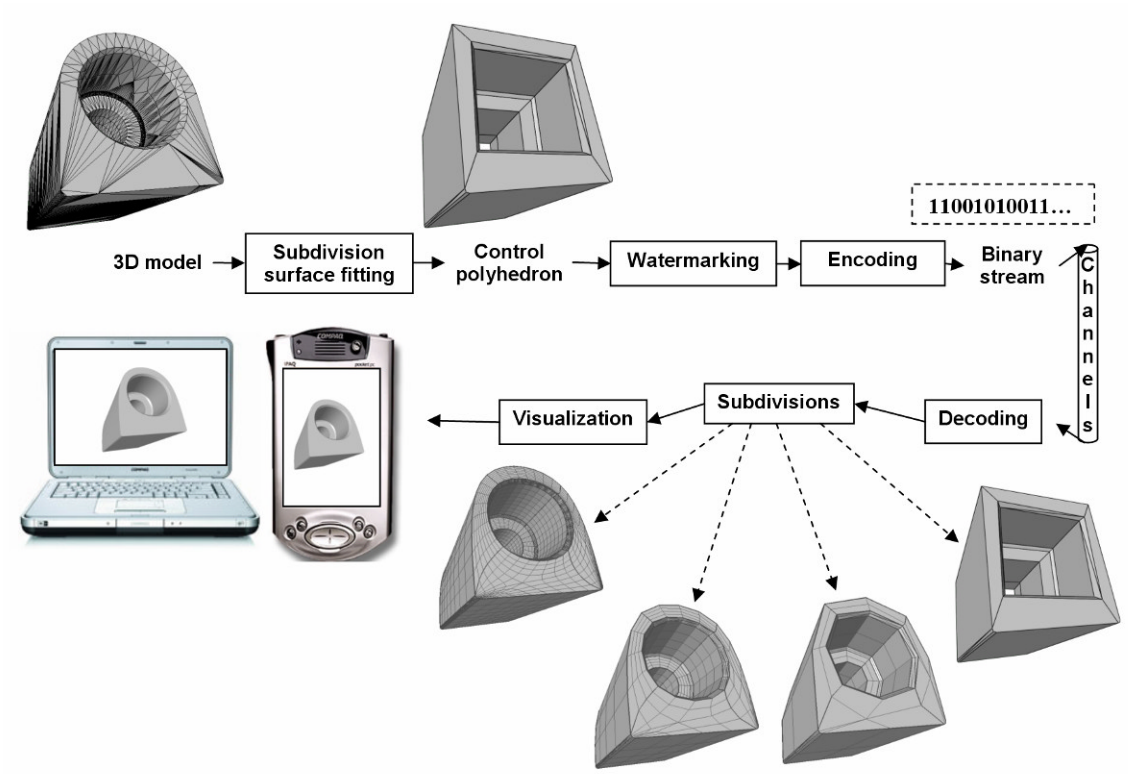


Figure 3. Complete application scheme.

IV. THE WATERMARKING ALGORITHM

A polygonal mesh is composed of geometry information which is a list of vertices (3D coordinates) and connectivity information which describes how vertices are connected to neighbors. The specificity linked to the compression method is the reduced number of vertices of the transmitted control polyhedron in which the mark will be inserted. On the other hand, this "light" model authorizes to also explore watermarking methods that are computationally expensive. Spectral analysis decomposition presents interesting properties in order to preserve fundamental characteristics of the shape of the control polyhedron.

A. Spectral Analysis

From connectivity, we compute a mesh Laplacian proposed by Bollabás [17]. Its formulation differs from what Karni and Gostman have proposed in [4] and is also known as *Combinatorial Laplacian* or *Kirchhoff Matrix*. This matrix is defined by

$$K = D - A \qquad (1)$$

where D is a Diagonal Matrix whose diagonal element D_{ii} corresponds to the valence of the vertex i (the valence is equal to the number of edges issued from this vertex) and A is the adjacency matrix of the mesh whose each element is defined by

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

For a mesh with n vertices, the matrices A, D and K have a size $n\times n$. The eigenvalue decomposition of the Kirchhoff Matrix K gives n eigenvalues l_i and n eigenvectors w_i. By sorting the eigenvalues in an ascending order, the n corresponding eigenvectors form a basis of 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.

Let X, Y and Z be three vectors containing the geometry information of the mesh whose vertices are $v_i=(x_i,y_i,z_i)$:

$$X=(x_1,x_2,\ldots x_n), \quad Y=(y_1,y_2,\ldots y_n), \quad Z=(z_1,z_2,\ldots z_n). \qquad (3)$$

The spectral decomposition is obtained by projection of these three vectors on the eigenvector basis and gives Xs, Ys and Zs:

$$\begin{cases} Xs = WX \\ Ys = WY \\ Zs = WZ \end{cases} \qquad (4)$$

The geometry can be restituted from spectral coordinates and the inverse matrix W^{-1}:

$$\begin{cases} X = W^{-1} Xs \\ Y = W^{-1} Ys \\ Z = W^{-1} Zs \end{cases} \qquad (5)$$

The amplitude spectrum can be obtained by computing coefficients S_i for each vertex by using the transformed coordinates with the following equation:

$$S_i = (||Xs||^2 + ||Ys||^2 + ||Zs||^2)^{½} \qquad (6)$$

Figure 4 presents the spectrum obtained with the *Bunny* model with 502 vertices and shows a very fast decrease after the second coefficient. The mark insertion process will take into account this amplitude value.

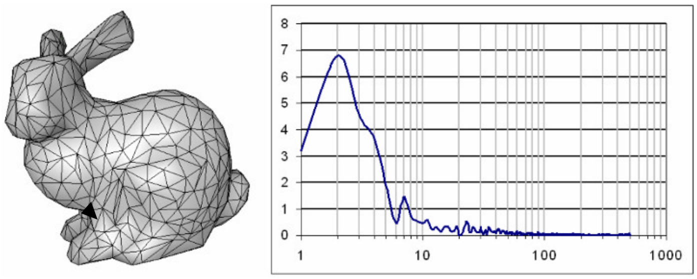
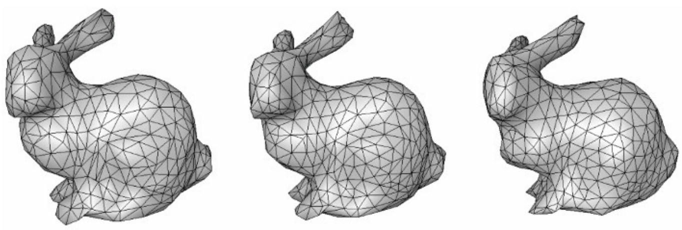


Figure 4. *Bunny* model and its spectral decomposition (S_i for $i=1$ to $i=502$).

Figure 5 shows the bunny model after reconstruction for different numbers of spectral coefficients. It appears that the first low frequency coefficients define roughly the shape, which is progressively refined as higher frequency coefficients are added.



250 spectral coordinates 150 spectral coordinates 50 spectral coordinates

Figure 5. Spectral reconstruction of *Bunny* model (502 vertices) with different number of spectral coordinates.

B. *Message sequence generation*

The inserted sequence is generated from a secret key and the initial message to hide. Depending on its length l, the message is spread out by a factor r:

$$r = \left\lfloor \frac{3n}{l} \right\rfloor \qquad (7)$$

i.e. each bit of the message is repeated r times leading to a sequence of length $r \times l$. The key initializes a pseudo-random generator to produce a pseudo-random sequence whose length also equals $r \times l$. Then, the pseudo-random sequence and the spread message are combined with a XOR operator. At last, the resulting binary sequence B is added to the coordinate data.

C. *Watermarking scheme*

Like Ohbuchi et al. [18], we modulate the object's coordinates in the transformed domain with the message to be inserted. For each Xs, Ys, or Zs coordinate, one bit can be embedded, so for each vertex, 3 bits can be dissimulated. Two different modulations have been retained.

Method 1:

Let Cs_i denote a mesh spectral coefficient, i.e. the value of either Xs, Ys or Zs for the ith vertex.

Cs_i' denotes the equivalent coefficient i.e. the value of either Xs', Ys' or Zs' for the ith vertex after watermarking. Cs_i' is computed as follows:

$$Cs_i' = Cs_i\,(1 + a b_i) \qquad (8)$$

where a is the strength of the watermarking;
b_i is the ith bit of the message sequence B to be embedded
($b_i = -1$ if sequence = 0 and $b_i = 1$ if sequence = 1).

Method 2:

Another scheme has been tested for insertion which improves the robustness of the watermark by increasing the mark strength for small coefficients. Indeed, as the amplitude spectrum decreases very rapidly, the quantities $Cs_i a b_i$ modifying the mesh coordinates become quickly negligible. To avoid this, small coefficients are modified additively, in the following way:

$$\begin{array}{lll} \text{if} & Cs_i < T & Cs_i' = Cs_i + T a b_i \\ \text{else} & & Cs_i' = Cs_i\,(1 + a b_i) \end{array} \qquad (9)$$

where T is a predefined threshold.

The modified object coordinates X', Y', Z' are then returned from Xs', Ys' and Zs' using equation (5).

D. *Mark Extraction*

The extraction operates in a non-blind manner (the initial unwatermarked object is necessary) on the same principle as the insertion: the spectral decomposition is applied to the initial mesh and to the watermarked mesh simultaneously. The Xs,Ys and Zs coefficients are then compared with the Xs'', Ys'', Zs'' obtained from the watermarked mesh and a message sequence B'' is generated depending on the sign of the differences Xs-Xs'', Ys-Ys'', Zs-Zs''. Note that Xs'', Ys'', Zs'' may differ from Xs', Ys', Zs', since the watermark mesh may have been modified by various processing (re-meshing, noise addition, voluntary attacks…). For ownership proof, the extracted sequence B'' is finally correlated with the inserted one (B) to verify if the mark is present or not. For message extraction, the B'' sequence has to be combined with the pseudo-random sequence generated with the secret key using a XOR operator, to give the spread message.

From what precedes, it is clear that the initial object and the watermarked one must have the same connectivity, in order to have the same eigenvalues and eigenvectors. Re-meshing and simplification operations will then make it impossible to extract the mark until the object is re-meshed identically to the initial one. So, before extraction, the connectivity of the watermarked object is verified and if different from the initial object, new vertices are computed from projection of the initial object vertices onto the watermarked object surface.

Other problems which may occur are objects misalignment or rotations that are avoided with an initial registration.

E. Watermarking of compressed models

The watermarking scheme described above has been used to protect the control polyhedrons of compressed models. If the insertion step remains the same, the extracting procedure depends on the nature of the watermarked model to process.

The copyright proof must, indeed, be brought on models intercepted to different levels (Figure 6):

a. When the object is transmitted on the channel: in this case, the hacker must possess the decoding algorithm. Moreover, at this level, the data can also be protected by encryption;
b. After decoding, by capturing the compressed objects before visualization;
c. After subdivision, by intercepting the final object for example from OpenGL stream during display.

In the two first cases, the mark must be extracted directly from the watermarked control polyhedron, in a similar manner as above.

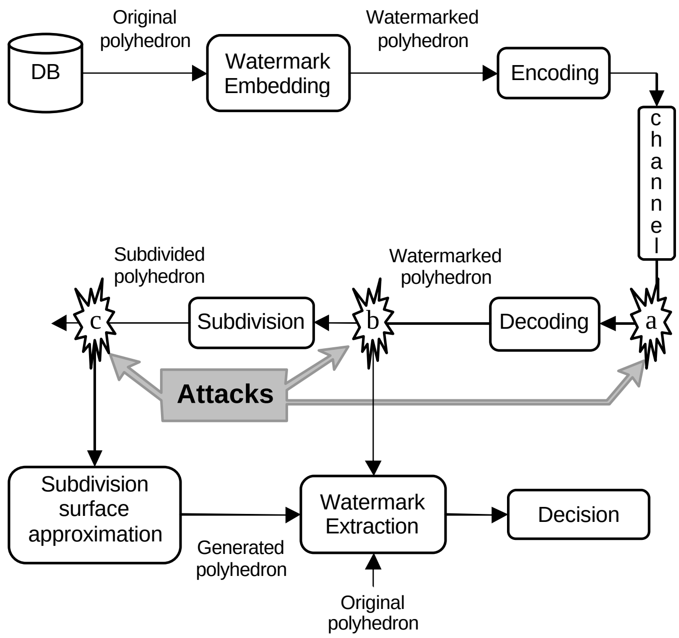


Figure 6. Mark detection process after three possible model attacks

In the third case, only a subdivided model is available, so that the control polyhedron is unknown. Hence, the segmentation algorithm is applied on the subdivided model, followed by the contours and surfaces approximation and leading to a new control polyhedron describing the watermarked object. Then, the detection algorithm can be applied. In the results section, figure 9 illustrates the whole detection process.

V. EXPERIMENTS AND RESULTS

The proposed method has been applied on different objects coming from CAD, for various values of the mark strength, and different attacks. Some results are presented hereafter. Before processing, all objects coordinates have been normalized in a unit bounding box.

A. Invisibility

An example of results for the Fandisk object is shown on Figure 7, for the insertion methods described by equations (8) and (9), with different values of α coefficient. The compressed model (control polyhedron) is composed of 75 vertices and the hidden message length is 64 bits, which allows a redundancy ratio of 3. For the same object, Table I gives the corresponding errors values L_1, L_2 and L_{max} computed from the difference between the original subdivided model and the watermarked subdivided model. Two examples are given for α=0.01 and α=0.1. The maximal error is found less than 1 percent when $a = 0.01$ and less than 6 percents when a reaches 0.1. This upper bound is deliberately high to visually reveal the model deformation, but will never be used for practical applications. For all other tests the values of a have been kept below 0.01.

From a qualitative point of view, the watermarked objects present no visual distortion unless the mark strength becomes important ($a \geq 0.05$). The second method gives similar results in terms of invisibility, although for equal mark strengths, the inserted signal energy is larger.

TABLE I. ERRORS MEASURED BETWEEN SUBDIVIDED WATERMARKED AND SUBDIVIDED ORIGINAL FANDISK MODEL

	Errors[a]			
	METHOD 1 a **= 0.01**	***METHOD 1*** a **= 0.1**	***METHOD 2*** a **= 0.01** **T= 0.05**	***METHOD 2*** a **= 0.1** **T= 0.05**
L_1 error	0,001420	0,012221	0,001416	0,012233
L_2 error	0,002032	0,016934	0,002032	0,016814
Max error	0,006748	0,057888	0,006868	0,056788

a. Objects are normalized in a unit bounding box

B. Robustness against attacks

Robustness has been tested for two plausible attacks:

- noise addition on the control polyhedron coordinates;
- re-meshing operations on the subdivided model.

Noise attacks have been carried out for various values of the watermark strength and insertion method 1 and 2. For method 2, T has empirically been chosen equal to 0.05.

The watermark is correctly found even in the presence of additive noise. The curves on Figure 8 represent the values of the correlation coefficient between the initial and the extracted message, when the watermarked model is distorted by addition of uniform noise and good detection (correlation coefficient above 0.5) is obtained until the noise level reaches approximately 0.002 for method 1 and 0.004 for method 2. This seems a good result, bearing in mind the small number of vertices and the reduced redundancy. These results also prove the superiority of method 2.

Re-meshing attacks have been simulated as illustrated on figure 9. The compressed object shown as example is composed of 24 vertices, and a 36 bits message has been inserted with a redundancy of 2. a has been set to 0.01 and T to 0.05. At first, the object has been watermarked, then subdivided and finally it has been remeshed (the object vertices and connectivity have been modified).

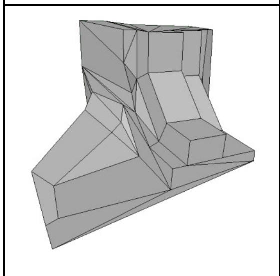
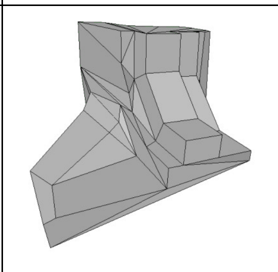
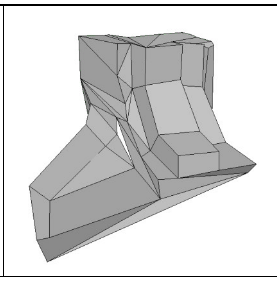
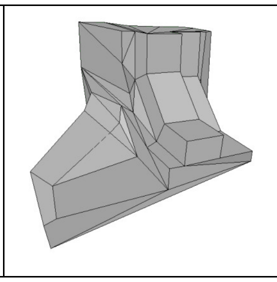
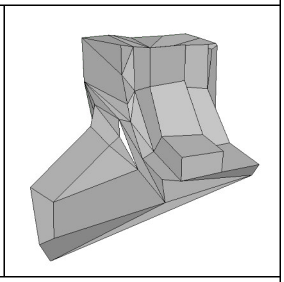
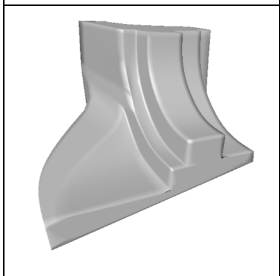
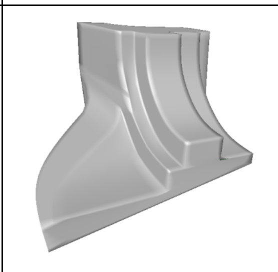
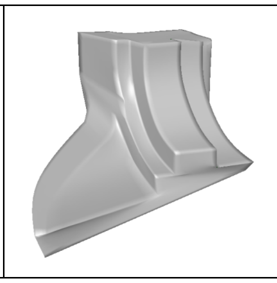
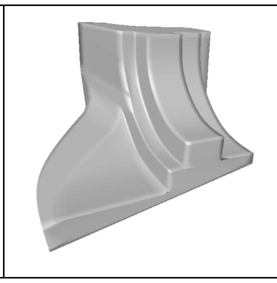
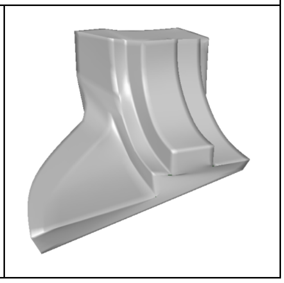
	Method 1 $a = 0.01$	Method 1 $a = 0.1$	Method 2 $a = 0.01$, $T = 0.05$	Method 2 $a = 0.1$, $T = 0.05$
Control polyhedron	Watermarked control polyhedrons			
				
Subdivided model	Watermarked subdivided models			
				

Figure 7. Control polyhedron and subdivided model without watermaking (left column) and with watermarking for the two methods with two different α values applied on the Fandisk model (4 right columns)

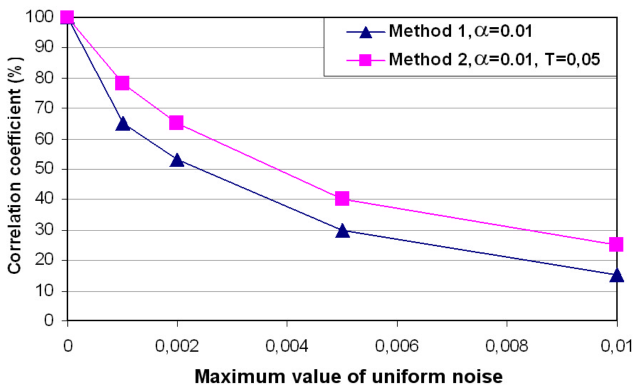


Figure 8. Correlation coefficient as a function of uniform noise level for Methods 1 and 2 with two different a values measured on Fandisk object.

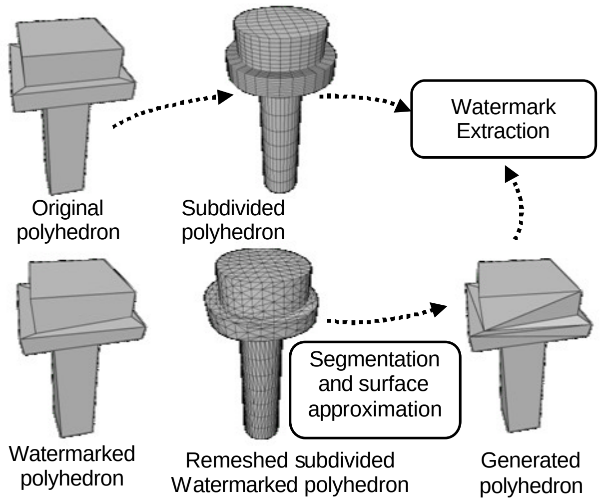


Figure 9. Mark extraction process for a subdivided polyhedron attacked by re-meshing

The detection has been achieved after a new step of segmentation and subdivision surface approximation. The new control polyhedron reveals the same number of vertices but a different connectivity from the original model. Therefore, for signature extraction, the original object connectivity is considered and each vertex is identified with the nearest one in the original mesh. For the example of figure 9, the correlation coefficient between the original inserted message and the extracted one equals 0.89 and hence shows good robustness.

VI. CONCLUSION AND PERSPECTIVES

In this paper, we have considered a watermarking method in the entire process of 3d models transmission. We have presented the compression stage that can be seen as a very severe attack for the watermarking stage. For a real application, it is very relevant to verify that watermarking techniques support this kind of attack before an eventual real attack. We have chosen a geometrical watermarking approach in the compression domain, but the main difficulty is the very low number of vertices. The developed method is non-blind as a consequence of the very compact compressed objects processed which extremely limits redundancy.

The results obtained have shown the robustness of the developed method against noise attacks, subdivision and re-meshing. The mark invisibility has been shown in spite of subdivision occurring after watermarking. As a conclusion, for the challenging aim of compact models watermarking, we find these results very encouraging.

In a future work, we plan to improve the message coding technique to increase the robustness against attacks. Other points of focus will be advances in the extraction process to offer better resistance when objects have been re-meshed or distorted with noise. In this case, the mark extraction implies to recover a control polyhedron whose connectivity is the same as the original object one. Presently, a segmentation and surface approximation is applied, but the segmentation is sensitive to noise. A probably better approach that will be tested in the near future consists in iterative modifications of the original control polyhedron (control points displacements) till the corresponding subdivided object reaches the watermarked one. Furthermore, this will avoid connectivity problems.

At least, more systematic tests have to be implemented on a larger object database, which has not been done yet in this first preliminary work.

ACKNOWLEDGMENT

This work is partially supported by the French Research Ministry and the RNRT (Réseau National de Recherche en Télécommunications) within the framework of the Semantic-3D national project (http://www.semantic-3d.net). We thank Helena Olmos and Sergio Espinosa for their development contribution.

REFERENCES

[1] M. Isenburg and J. Snoeyink, "Face Fixer : Compressing Polygon Meshes with Properties", ACM Siggraph Conference Proc., 2001, pp. 263-270.

[2] C. Touma and C. Gotsman, "Triangle mesh compression", Graphic Interface Conference Proceedings, 1998, pp. 26–34.

[3] A. Khodakovsky, P. schroder and W. Sweldens, "Progressive Geometry Compression", ACM Siggraph Conference Proc., 2000, pp. 271-278.

[4] Z. Karni and C. Gotsman, "Spectral compression of mesh geometry", ACM Siggraph Conference Proc., 2000, pp. 279-286.

[5] R. Ohbuchi, H. Masuda, and M. Aono, Watermarking ThreeDimensional Polygonal Models, Proceedings of the ACM Multimedia '97, Seattle, Washington, USA, November 1997, pp. 261-272.

[6] Kanai, S., Date, H. and Kishinami,T., Digital Watermarking for 3D Polygons Using Multiresolution Wavelet Decomposition, Proceedings of the Sixth IFIP WG 5.2 International Workshop on Geometric Modeling: Fundamentals and Applications (GEO-6), pp. 296-307, Tokyo, Japan, December 1998.

[7] K. Yin, Z. Pan, J. Shi, D. Zhang, Robust mesh watermarking based on multiresolution processing, Computers & graphics, 25, 2001, pp. 409-420.

[8] E. Praun, H. Hoppe, A. Finkelstein, Robust mesh watermarking, ACM SIGGRAPH 1999, pp. 69-76.

[9] R. Ohbuchi, S. Takahashi, T. Miyazawa, and A. Mukaiyama, "Watermarking 3D polygonal meshes in the mesh spectral domain," in Proc. of Graphics Interface, Ottawa, Canada, June 2001, pp. 9-17.

[10] Li, L., Zhang, D., Pan, Z., Shi, J., Zhou, K., Ye, K., Watermarking 3D mesh by spherical parameterization, Computers and Graphics (Pergamon) 28 (6), pp. 981-989, 2004.

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

[12] E. Catmull and J. Clark, "Recursively generated b-spline surfaces on arbitrary topological meshes", Computer-Aided Design, vol. 10, no 6, 1978, pp. 350-355.

[13] J. Stam and C. Loop, "Quad/triangle subdivision", Computer Graphics Forum, vol. 22, no 1, 2003, pp. 79-85.

[14] G. Lavoué, F. Dupont, A. Baskurt, "Subdivision surface fitting for efficient compression and coding of 3D models", SPIE Visual Communications and Image Processing, 2005.

[15] G. Lavoué, F. Dupont, A. Baskurt, "A new CAD mesh segmentation method, based on curvature tensor analysis", Computer-Aided Design, Vol. 37, No. 10, 2005, pp. 975-987.

[16] G. Lavoué, F. Dupont, A. Baskurt, "A new subdivision based approach for piecewise smooth approximation of 3D polygonal curves", Pattern Recognition, Vol. 38, No.8, 2005, pp. 1139-1151.

[17] Bollabás, Modern graph theory, Springer, 1998.

[18] Ohbuchi, R., A. Mukaiyama, and S. Takahashi, "A frequency-domain approach to watermarking 3D shapes", Computer Graphics Forum, 2002, 21(3), pp. 373-382.