Spread spectrum robust watermarking for NURBS surfaces

MIHAI MITREA1,2, TITUS ZAHARIA1, FRANCOISE PRETEUX1
1 ARTEMIS Project Unit, GET / INT, Evry-France
2 Faculty of Electronics and Telecommunications, POLITEHNICA University of Bucharest-Romania
9, Rue Charles Fourier, 91011 Evry
FRANCE
mihai.mitrea@int-evry.fr http ://www-artemis.int-evry.fr

ABSTRACT

This paper presents a robust watermarking method for 3D data represented as NURBS. The mark is generated starting from a public message on 64 bits and from a secret information on 22 bits. The mark is embedded into the Discrete Cosine Transform (DCT) coefficients corresponding to the control points, by means of a spread spectrum technique. The detection procedure does not require the original (unmarked) data. The proposed method features transparency and robustness with respect to knot insertion/removal, degree elevation/reduction, geometric attacks, compression, linear and non-linear filtering, noise addition and multiplication. The probability of false alarm is lower than $16^{-14}$.

Keywords: NURBS representations for 3D data, DCT coefficients, robust watermarking, spread spectrum.

1. INTRODUCTION

Emerging from a two thousand year tradition of military and diplomatic applications (well-mentioned in the literature, from Herodotus’ Histories up to Shannon's works) the watermarking has been recently identified as the unified framework under which a large range of problems directly connected to digital data distribution over Internet can be solved [1-3]: property right identification, forgery tracking-down and content authentication. Such issues are of an utmost interest for various fields. For instance, the film producers search a possibility for distributing their products with a strict control of copies. Similar problems should be solved by the 3D character creators or by companies which are obliged to distribute their CAD (Computer Aided Design) objects over insecure networks. The 3D data are represented either as meshes or as NURBS - Non Uniform Rational B-Spline. This paper presents a new watermarking method which ensures the property rights protection for NURBS surfaces.

1.1. The watermarking framework

Let us take the example of a digital 3D product which is Alice's property. When Bob buys this product, he can copy it and he might try to sell it again to somebody else, e.g. to Carol. Of course, this situation can be repeated as many times as there is an interest in the considered 3D data: note that in the digital world the copies are exact (without any loss of quality). In such a scenario, although Alice may be aware of the unauthorised distributions of her data, she cannot do anything in order to protect her property rights. First, Carol can just pretend that the 3D model is her property. Secondly, Bob can decline any involvement in such a distribution. Therefore, Bob may try to process the marked data she bought in order to turn the Alice mark undetectable. Such a transform applied by Carol is referred to as attack. Of course, Bob is also
interested in attacking Alice's mark, in order to be able to sell the copies he can make. A watermarking procedure which can face various attacks is referred to as robust. On the other hand, the main issue of fragile watermarking deals with is data authenticity: any alteration made into a fragile marked object should be detectable. For instance, such a method would be useful for forensic applications, for banknote issuing or for cheque printing.

As compared to watermarking, steganography is a related yet different field. It refers only to transparent data embedding and assumes that no attack would be encountered. This is the case when the malicious users do not know the very existence of the hidden information (e.g. Carol does not know that Alice protected her data). There are also another application of steganography: text insertion into a video file for a quick video indexing, a time information into a fixed format image file, etc.

From the communication theory point of view, a watermarking procedure may be modelled as a noisy channel [1-4]. The Alice mark stands for the message to be transmitted. The original (unmarked) 3D data and the attacks play the role of the noise (they alter the information during the transmission). On the other hand, for the human observer, the embedding procedure a priori leads to some artefacts in the marked video. Hence, the transparency requirement means a very low power for the mark. It is known [1-3] that the spread spectrum - SS - modulation techniques provide the best framework for low power signal transmission. According to the targeted application, a trade-off between transparency and robustness should be reached.

Concerning still images, video and speech, a large number of research studies addressed the challenging issue of such a trade-off and are reported in literature [1-10]. However, there are quite few publications in this respect when dealing with 3D data [11-21], despite the huge demand for their protection by means of watermarking techniques. On the one hand, there is not a direct extension of the 2D techniques toward the geometric data [18]. On the other hand, the robustness problem to be addressed when dealing with 3D data is more complex [18]. All these peculiarities will be discussed in the next section.

1.2. The 3D watermarking peculiarities

We shall further present the main principles involved in the 3D watermarking. At a first glance, there are two classes of methods which correspond to two different ways in which the 3D data are represented: meshes and NURBS.

The largest part (more than 90%) of the studies on 3D data watermarking were mesh-based, no matter they were devoted to fragile watermarking [11] or to robust watermarking [12-19]. Generally, the mesh-based techniques embed the mark into the vertex positions. Both non-oblivious and oblivious methods have been reported. However, the main problem remains the robustness: in this respect, practically all the above-mentioned methods have some weak points.

For instance, in [13] a set of scalar basis functions over mesh vertices is built up in order to overpass the lack of a natural parameterisation for frequency-based decomposition. The experimental results refers only to relative smooth surfaces and the detection can not be achieved without information concerning the original mesh connectivity (the method is non-oblivious).

The work in [15] is focussed on deriving a selection criterion (based on a distance measure) which ensures the lowest visibility. The method thus obtained is oblivious and robust against scaling, rotation or any combination of global geometrical transforms. No result concerning compression robustness is reported.

In order to achieve the same robustness against geometric transformations, [16] considers the principal component analysis. Here again, no reference concerning the robustness against compression is met.

Alternatively, in [20] and [21], 3D data are represented as NURBS [20], [22] surfaces. A NURBS surface is defined by means of [22]:

- a set of control points which should be approximated by the surface;
- two knot vectors which determine the influence of the control points on the surface, and
- a set of weights which somehow establishes how close the surface is to the control points.

In order to watermark the 3D model, the principle in [20] is to change the weight and the knot vectors so as to preserve the overall geometry. Such an algorithm is required mainly when marking CAD models, where the smallest alteration of the geometry would lead to errors in production. The main drawback of this method is its lack of robustness: the weakest attack can turn the mark undetectable. Hence, such an approach rather belongs to steganography than to watermarking per-se.

However, when dealing with NURBS representations, the main advantage is the availability of a natural 2D surface parameterisation which makes it possible to extend the 2D watermarking techniques to 3D data. As far as we know, [21] is the first study which exploits the NURBS parametric definition in this respect: the authors advanced a method where the mark is not embedded directly into the 3D model but into some virtual images derived from the 3D model. They propose two methods, the first for steganography and the second for watermarking. For the latter method, three virtual images are obtained by uniformly sampling the 3D model and by recording the x, y, and z co-ordinates thus obtained. The method was applied to 2 models (sampled as 128×128 and 64×64 points) and proved itself robust against control point modification by affine transformations, knot vector modification, and surface approximation. The probability of false alarm was evaluated as being lower than 10⁻⁷ (which can be disturbing for some applications). Moreover, the method performances seem to depend on the 3D model size. Finally, this method is computational complex.

In conclusion, at our best knowledge, there is no 3D watermarking method which meets the obliviousness, robustness and low false alarm probability.

Section 2 presents the method we developed in order to reach the trade-off among these requirements. In
contrast to the method in [21], the mark is embedded into some images corresponding to the control points. The robustness (with respect to both NURBS-based and general attacks) and the low probability of false alarm (lower than $16^{-14}$) are derived from the spread spectrum embedding procedure we developed. The experimental results are presented in Section 3: they correspond to about 5000 3D models, with different content (CAD models, cartoon characters, and game characters) and with different model parameters (different number of knots and of controls points). Section 4 concludes the paper and open the perspectives for our future work.

2. THE DEVELOPED METHOD

The approach we developed is structured into four main steps. First, 3 virtual images are computed starting from the NURBS representation of the original 3D model (Figure 1). The 2D-DCT is individually applied to each image thus obtained and a vector of salient characteristics is selected. Second, the secret information (the key) and public information (the logo) are combined by means of a Code Division Multiple Access (CDMA) technique, thus providing the watermark. Third, the watermark is embedded into the salient characteristic vector. Finally, the mark is obviously detected by matched filters.

2.1 From 3D data to a characteristic vector

This section aims at extracting from the NURBS representation of the 3D object some salient characteristics which should be able to convey the mark. For clarity sake, the main definitions related to NURBS representations are recalled [22].

Nowadays, NURBS may be considered as a de facto standard for representing, designing and data exchanging framework, NURBS makes it possible both analytic functions defined on the $U$ and $V$ knot vectors, according to (2):

$$N_{i,p}(u) = \begin{cases} 1 & u_i \leq u \leq u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

Equation (2) expresses that the knot vectors determine the influence area of a $P_{i,j}$ control point (the area in the $(u,v)$ space where $N_{i,p}(u) \cdot N_{j,q}(v) \neq 0$).

Several NURBS representations can correspond to the same object [22]. In other words, there are several $p$, $q$, $U$, $V$, $n$, $m$, $w_{i,j}$, and $P_{i,j}$ entities which led to the same $S(u,v)$ surface.

Starting from the NURBS representation of the 3D object, three virtual images (matrices) are built up (Figure 1): the pixel values in these images are the result of an embedding procedure we developed. The experimental results are presented in Section 3: they correspond to the three virtual images. Be the $v$ the component, the closer the NURBS surface to the corresponding control point.

- $U$ and $V$ are two knot vectors (nondecreasing sequences of real numbers):

$$U = \left[0, 0, ..., 0, p + 1, u_{p+2}, ..., u_{r-1}, 1, ..., 1\right]$$

$$V = \left[0, 0, ..., 0, q + 1, v_{q+2}, ..., v_{s-1}, 1, ..., 1\right]$$

where $r = n + p + 1$ and $s = m + q + 1$.

- $N_{i,p}(u)$, $i \in [0, ..., n]$, and $N_{j,q}(v)$, $j \in [0, ..., m]$ are the $p$th and $q$th degree nonrational B-spline basis functions defined on the $U$ and $V$ knot vectors, respectively.

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u)$$

Equation (2) expresses that the knot vectors determine the influence area of a $P_{i,j}$ control point (the area in the $(u,v)$ space where $N_{i,p}(u) \cdot N_{j,q}(v) \neq 0$).

Several NURBS representations can correspond to the same object [22]. In other words, there are several $p$, $q$, $U$, $V$, $n$, $m$, $w_{i,j}$, and $P_{i,j}$ entities which led to the same $S(u,v)$ surface.

Starting from the NURBS representation of the 3D object, three virtual images (matrices) are built up (Figure 1): the pixel values in these images are the $P^x_{i,j}$, the $P^y_{i,j}$, and the $P^z_{i,j}$ co-ordinates of the control points, respectively.

The 2D-DCT (2D Discrete Cosine Transform) is applied to each image and the coefficients thus obtained are sorted in a decreasing order.

Finding the right place for the mark embedding is a crucial issue. On the one hand, when embedding the mark in the upper ranks of this hierarchy, some artefacts might occur in the marked object. On the other hand, when embedding the mark in the lowest ranks of this hierarchy, a weak robustness is expected. Hence, in order to reach the trade-off between robustness and transparency, the mark should be embedded into some middle rank coefficients. Be there the $M$ coefficients having the $\rho$ ranks, where $\rho_{inf,n \times m} \leq \rho \leq \rho_{sup,n \times m}$; hence $M = \rho_{sup,n \times m} - \rho_{inf,n \times m} + 1$. We emphasise that the rank limits depend on the image size, i.e. on the $(n+1) \times (m+1)$ number of control points. A 3D object can provide a characteristic vector having $N = M \times 3$ components, obtained by concatenating the coefficients corresponding to the three virtual images. Be $\nu$ the
vector of these coefficients and be $\lambda$ the vector of the corresponding spatial frequencies. There is no constraint concerning the order of the coefficients in the $\nu$ vector. We considered first the coefficients corresponding to the $P_{i,j}^x$, then to the $P_{i,j}^y$ and finally to the $P_{i,j}^z$. Of course, the coefficients corresponding to the three virtual images can be also interlaced, according to their rank.

$$P_{i,j} = [P_{i,j}^x, P_{i,j}^y, P_{i,j}^z]$$

![Figure 1: From 3D data to a characteristic information: the DCT coefficients of the $P_{i,j}^x$, $P_{i,j}^y$, $P_{i,j}^z$ images are sorted in a decreasing order and some middle ranks are considered.](image)

2.2 Generating the mark

According to the general watermarking requirements [1-3], [5], the public information, i.e. the information Alice wants to insert into her 3D data, is represented on 64 bits (16 digits in hexadecimal). Be these hexadecimal digits denoted by $s_1,s_2,\ldots,s_{16}$. This message is modulated by means of an SS-CDMA technique, according to the suggestions in [7], [9], [10].

This procedure starts by considering 16 bipolar $(-1/1)$ pseudo-noise sequences (one sequence for each hexadecimal digit in the message) of an $N+15$ length. These sequences, denoted by $[n_i]_{i \in \{1,\ldots,16\}}$, should be orthogonal so as to afford an optimal detection: the performances of any SS technique depend upon the quality of the pseudo-random number generator it involves in, see [9], [10], [23]. In our experiments, we considered a generator implemented by means of an LFSR - Linear Feedback Shift Register. For such a generator characterised by a $d$ degree primitive polynomial over GF(2) - the Galois Field of second order, the output sequence orthogonality is theoretically proved [24]. Other generators may be considered, as those based on the Gold codes [23] for instance.

The output of an LFSR generator takes the 0 and 1 values. In order to obtain the $-1/1$ bipolar values, the 0 value is just replaced by $-1$. As the output of a primitive LFSR has a $2^d-1$ period, the condition to be fulfilled is $2^d-1 \geq 16(N+15)$. We shall further consider just one period from such a generator and we shall cut from it the 16 $n_i$ sequences of $N+15$ length.

The $d+1$ coefficients of the primitive polynomial stand for the key in our watermarking method (the secret information Bob and Carol should know).

Further on, each $s_i$ symbol, $i \in \{1,\ldots,16\}$, is mapped to an $r_i$ sequence. These 16 $r_i$ sequences have an $N=M \times 3$ length and are extracted from the corresponding $n_i$ sequence, according to (3):

$$s_i \leftrightarrow r_i = [n_i, n_i+1,\ldots, n_i+N-1]$$

As the $s_i$ symbol can take a value ranging from 0 to 15, the $n_i$ sequences should have the above-mentioned $N+15$ length in order to allow the $r_i$ sequences to be obtained.

2.3 Embedding the mark

The mark to be embedded is a vector denoted by $x$ which is the sum of the 16 $r_i$ vectors, $i \in \{1,\ldots,16\}$, (4); hence, the $x$ mark has the same length as any $r_i$ vector:

$$x_j = \sum_{i=1}^{16} n_{i,j}, j \in \{0,N-1\}$$

where $x_j$ represents the $j$ index component of the $x$ vector, while $r_{i,j}$ stands for the $j$ index component of the $r_i$ vector.

This mark is embedded into the selected $\nu$ coefficients (Figure 1) by means of a weighted addition [8-10]. The embedding procedure is described by (5):

$$\nu_j = \nu_j \cdot (1 + \sigma \cdot x_j)$$

where the lower index denotes the component of the respective vector, $j \in \{0,\ldots,N-1\}, \nu$ denotes the marked coefficients while $\sigma$ stands for a constant value. This multiplication by a $\sigma$ constant adjusts the power of the $x$ mark. The larger the $\sigma$ value, the greater the robustness but the worst the transparency.

Further, the $\nu$ original coefficients are exchanged for the $\nu'$ marked coefficients and three 2D Inverse DCT are computed, thus obtaining the marked virtual images.

Finally, an affine mapping is individually applied to each marked virtual image in order to preserve the original minimal and maximal values, thus obtaining the marked control points.

The watermarking procedure is achieved: the marked object is represented by the original knot vectors, by the original weights, and by the marked control points.

2.4 Detecting the mark

Let us suppose that Alice finds a 3D object she thinks is her in Carol's possession. In order to prove her property rights on these copies, Alice computes the 2D-DCTs on all the corresponding virtual images. Carol's sequence generally differs from the marked sequence Alice sold to Bob. This difference has several reasons:
Bob might try to attack the watermark or Carol just wanted a compressed version of the 3D objects. In any situation, Alice records the coefficients which correspond to the locations (Figure 1); the vector obtained by concatenating these coefficients is denoted by \( \nu' \).

The public message is recovered \([7],[9],[10]\) by means of \( R_{\nu''n_j} (\cdot) \) cross-correlation functions, computed between \( \nu'' \) and each \( n_i \), \( i \in \{1,...,16\} \) sub-sequence. There is no need for these \( n_i \) sequences to be recorded: Alice may simply compute them again, starting from the polynomial coefficients.

The peak position in such a cross-correlation functions represents the \( \hat{s}_i \) recovered symbol. That is, the \( \hat{s}_i \) recovered symbol is implicitly involved in (6):

\[
R_{\nu''n_j} (\hat{s}_i) = \max R_{\nu''n_j} (t), \ t = \{0,1,...,15\} . \quad (6)
\]

Although these cross-correlation functions are not delta-like because of the noise and of the attacks, the peak position might keep its original position, as a consequence of the fact that the \( n_i, i \in \{1,...,16\} \) sequences were orthogonal. However, when the peak position change its position, an error is encountered.

### 3. EXPERIMENTAL RESULTS

Our method was applied to about 5000 NURBS models, with heterogeneous contents: human bodies, human environments, spare parts for car industry, etc. The related virtual images had very small sizes, e.g. \( 5 \times 4 \) pixels or \( 13 \times 20 \) pixels. Two examples are presented in Figure 2: Zozoo model (a virtual character for cartoons) and a CAD model (a part from the Renault’ Laguna car).

We shall first specify the numerical values for the method parameters. These values are heuristically found out in order to reach the general trade-off between transparency and robustness, while ensuring an oblivious detection and a small probability of false alarm. However, they can be adapted for any particular application. Then, we shall present the results concerning transparency and finally those concerning robustness.

#### 3.1 Instantiation of the method parameters

Under the Spread Spectrum framework, a long characteristic vector is required (i.e. the spectrum should be really spread). Hence, it may be considered that the most challenging issue is to mark small objects (i.e. objects where there is not too much room for the mark to be embedded). In our experiments we also dealt with very small 3D objects (e.g. objects for which the virtual images have \( (n+1) \times (m+1) = 5 \times 4 \) pixels). In such a case, we considered the coefficients having \( \rho_{\text{inf},4 \times 3} = 8 \) and \( \rho_{\text{sup},4 \times 3} = 15 \) (just \( M = 8 \) coefficients for each image). Hence, the \( \nu \) characteristic vector has just \( N = 8 \times 3 \) components which is too small for a SS technique. In order to fulfill the SS assumption, we considered sequences of 1024 objects and we concatenated the coefficients having ranks \( 8 \leq \rho \leq 15 \) in the DCT hierarchy corresponding to each image sequence: \( N = 8 \times 3 \times 1024 \).

When dealing with a very large number of control points (e.g. \( 128 \times 128 \) control points per object) a single object can be successfully marked. In such a situation, we consider \( \rho_{\text{inf},128 \times 128} = 8 \) and \( \rho_{\text{sup},128 \times 128} = 8199 \).

The most frequent applications are somewhere in between these two cases: the 3D objects are represented as a set of different NURBS (e.g. 20 NURBS for one object), each NURBS having a quite small number of control points (e.g. \( 10 \times 10 \) control points). In such a case, a sequence of about 10 objects would be required while considering \( \rho_{\text{inf},9 \times 9} = 8 \) and \( \rho_{\text{sup},9 \times 9} = 47 \).

As a final remark concerning the rank selection, notice that the 3D objects in a sequence to be marked may have different sizes. In such a scenario, the user can choose either to consider an equal number of coefficients for each object (determined by the smallest sized object) or to adapt this coefficient number to each object size.

The LFSR we consider is characterised by a primitive polynomial of \( d = 21 \) degree, value which ensures \( 2^d - 1 \geq 16(N+15) \). On the other hand, the \( d + 1 = 22 \) coefficients represents the key. It is known from the cryptography theory that \( d = 21 \) offers a large enough space key space so as to avoid an attack by exhaustive searching. In other words, in order to destroy the mark, Carol cannot try all the primitive polynomials of \( d = 21 \) degrees, because it would take too much time (at least years).

Concerning the \( \sigma \) parameter in (5), we found out that a 1/64 mark power reaches the trade-off between the robustness and the transparency desiderata.

#### 3.2 Transparency and robustness in practice

Let us now present the results concerning transparency. In this respect, we considered 10 human observers, with different ages and professional backgrounds: 6 researchers working in the field, 3 teenagers (for on-line gaming 3D models) and one expert from Renault (for the 3D car industry models). They generally agreed that the method features transparency (Figure 2).
Concerning the robustness, our experiments were conducted in two stages. First, we checked up that the common NURBS operations cannot turn the mark undetectable. Then, we considered the attacks inherited from image/video watermarking and we showed that our method can face them.

We first considered knot insertion/removal, knot refinement, degree elevation/reduction, changing the control point order and affine transformations (translations, rotations, scaling, shares).

Knot insertion and its converse operation knot removal are one of the most important B-Spline algorithms [22]. They are involved in evaluating surface points and derivatives, subdividing surfaces and in interactive design.

Be there the \( S(u,v) \) NURBS surface from (1), of \( p \) and \( q \) degrees, defined on the \( U \) and \( V \) knot vectors. Knot insertion means to find out an \( \widetilde{S}(u,v) \) NURBS surface, of the same degrees, defined on the \( U \) and \( V \) vectors which is identical to the first surface, (7):

\[
\forall 0 \leq u,v \leq 1, \ S(u,v) = \widetilde{S}(u,v).
\]

The \( \widetilde{U} \) vector includes \( U \) and contains just one additional element. If \( U = [u_0,u_1,...,u_l] \), then

\[
\widetilde{U} = [u_0,u_1,...,\bar{u}_k = u_k,\bar{u}_{k+1} = v,\bar{u}_{k+2} = u_{k+1},...,\bar{u}_{l+1} = u_l]
\]

where \( \bar{u} \in [u_k,u_{k+1}) \) is the knot to be inserted.

The new control points can be computed starting from (7). In [22], it is proved that only \( p \) points differ from the original control points and that these new points are linear interpolations of some original control points.

For our watermarking application, this means that \( p \) columns have been inserted in the virtual images where the mark is searched for. Equivalently, a knot removal means a column removal operation.

In our experiments we inserted one knot in the \( U \) vector and one knot in the \( V \) vector and the mark was successfully detected. The same results was obtained when removing one knot from the \( U \) vector and one knot in the \( V \) vector.

Knot refinement means to insert many knots at once [22]. In such a situation, the mark could not be directly detected. In order to successfully recover the mark, such a transformation should be inverted. Hence, we make public the weights and the original knot vectors. Prior to the detection, the inserted knots can be removed and then the mark successfully detected. Notice that in such a scenario there is no contradiction with the obliviousness requirements: the mark is inserted only in original control points which are not required during the detection procedure. On the other hand, as the control points are salient characteristics for the 3D object, a malicious attacker would not succeed in forging the 3D points and to keep the knots and the weights unchanged.

As far as the 3D model thus obtained is still acceptable (from the human observer point of view) the mark is still detectable.

Degree elevation/reduction makes it possible to combine several NURBS surfaces into a single NURBS surface and to construct certain types of surfaces from a set a curves. For instance, such operations are required in cartoons [25]. By degree elevation, the new NURBS is identical to the original one but have an incremented order. When the detection is based only on the marked object, the method we developed is robust against degree elevation with one unity on both \( u \) and \( v \) direction (i.e. when considering a curve of \( (p+1) \)th and \( (q+1) \)th degrees). When disposing of the original knots and weights, and when inverting these operations, the robustness is achieved regardless the degrees at which the curve was elevated/reduced.

We also considered the scenario according to which the control points are randomly permuted. In the NURBS representation it is an inner relationship between the \( S(u,v) \) surface, the \( P_{i,j} \) control points and the \( U \) and \( V \) knot vectors: when changing the \( P_{i,j} \) control points, in order to keep the same \( S(u,v) \) surface, generally the knot vectors should also be changed. Hence, for such an attack, the \( U \) and \( V \) vectors should be public and used to invert the permutation. Moreover, notice that for 3D objects which have a symmetry, the control points can be circularly shifted without changing neither the surface nor the knot vectors. In such a situation, the control points should be pre-ordered (e.g. the first control point should be the one having the smallest \( P^x_{i,j} \) value).

The affine transformations (translations, rotations, scaling and shares) applied to a 3D object are, in fact, directly applied to the control points [22]. In other words, these are just affine transformations applied to the control points co-ordinate matrices. We experimentally verified that our method is robust against such operations.

At the second level, we considered several attacks inherited from image/video watermarking, namely: compression, noise addition/multiplication, linear and non-linear filtering and the Stirmark attack [26].

In order to compress the 3D object, we compressed the virtual images corresponding to the 3D objects, by means of the JPEG 2000 (codestream) algorithm. The ratio of the compressed file size to the uncompressed file size was 0.6. Although the compressed object is damaged, the mark was successfully detected.

The mark was also detected when the 3D models were corrupted with additive and multiplicative noise, with different probability density functions (uniform, Gaussian, gamma). We considered both white and coloured noise.

We checked up the resistance against Laplacian, Gaussian, and median filtering. These filters were individually applied to each virtual image with a window of 3 x 3.

Finally, we applied the Stirmark attack on each and every virtual image, and the mark was detected.

The probability of false alarm was evaluated as being lower than \( 2 \times 10^{-16} \approx 16^{-14} \).
4. CONCLUSION

In this paper, we present a watermarking method for 3D models represented as NURBS surfaces. The mark is embedded into the 2D-DCT coefficients by means of a spread spectrum procedure. The method features obliviousness, transparency, robustness with respect to the common attacks, and a probability of false alarm lower than 16−14. As a consequence of the generality of the embedding/detecting rule, there is no need for error correcting codes or other ready-to-use solutions which are known as increasing the watermarking method performances.

A future study will be devoted to the opportunity of applying a 3-D-DCT based method for watermarking.

ACKNOWLEDGEMENT

This research study was supported by the RNRT project SEMANTIC-3D: http://www.telecom.gouv.fr/rnrt.

REFERENCES