

PROTECTION OF 3D OBJECT VISUAL REPRESENTATIONS*

Jihane Bennour and Jean-Luc Dugelay

Institut Eurecom
2229 route des Cretes, B.P.193
06904 Sophia-Antipolis, France
Email: bennour@eurecom.fr, dugelay@eurecom.fr

ABSTRACT

In this paper, we describe a new framework for watermarking 3D objects via their contour information. Unlike most conventional 3D object watermarking techniques, for which both insertion and extraction of the mark are performed on the object itself (3D/3D approach), we propose an asymmetrical 3D/2D procedure. The goal of our work is to retrieve information (originally hidden in the apparent 3D silhouette of the object) from resulting images or videos having used the synthetic object, thus protecting the visual representations of the object.

After developing theoretical and practical key-points of this 3D object watermarking scheme, we present the results of some preliminary experiments.

1. INTRODUCTION

Real applications show that watermarking can be useful for several purposes. For example, viewers would like to check if the use of a given object is legal or not, to access additional information concerning the object (e.g for authentication or indexing), the owner (copyright), or even the buyer (e.g for non-repudiation).

These issues are taken seriously by many people and laboratories that work out solutions to be able to protect multimedia documents in general and 3D objects in particular. Several watermarking algorithms of 3D object have been proposed. Most of them are based on slight modifications performed on meshes via geometric and/or topological data of 3D objects. Typically, authors propose to modify either the 3D coordinates of some points or the connectivity of triangles within a mesh. Interested readers are invited to refer to the publications [2, 3, 4, 5, 6]. In almost all these published approaches, users are assumed to be able to access 3D datafile to extract/detect the mark. However, and based on our experience, it is usually more frequent to locate and recover suspect 2D images (resulting images or videos having used a synthetic

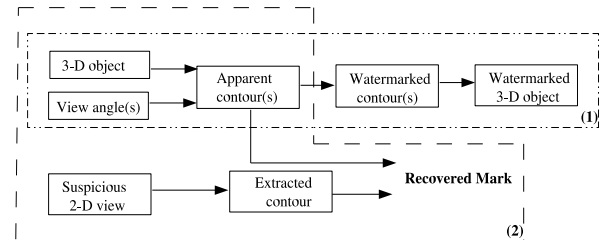


Fig. 1. Flowchart of our system: 1-Insertion; 2-Detection

object) than datafile of the 3D object itself. In this quite particular context, we can cite [7] which protect the 3D object usage through texture watermarking. We can then check if a 2D represented view is protected by recovering hidden data from watermarked texture. This algorithm assumes that 3D objects are realistic or at least rich in texture, which is not always the case.

The purpose of this article is to introduce a new framework for watermarking 3D objects based on the object's apparent contour instead of its texture map. Thanks to this point, it has the potential of protecting all the images derived (i.e. projected) from a 3D object with or without texture, after it has been watermarked in 3D (see figure 1).

After presenting the proposed framework in section 2, we will provide some details about algorithms in section 3 then we will present some preliminary results in section 4 and conclude with some possible future directions in section 5.

2. PROPOSED FRAMEWORK

Before discussing in details the implementation of our watermarking algorithm based on 3-D apparent contour and the results of some preliminary experiments, we first start by giving an overview of our approach.

As shown in figure 2, given a known 3D object consisting of a geometric definition (represented by 3D polygonal meshes) we can protect it by watermarking its apparent contour. We first extract the silhouette of the object to watermark (step 1), once the extracted silhouette is sampled (step 2) we water-

*THIS RESEARCH IS SUPPORTED IN PART BY THE FRENCH NATIONAL RNRT SEMANTIC 3D PROJECT [1]

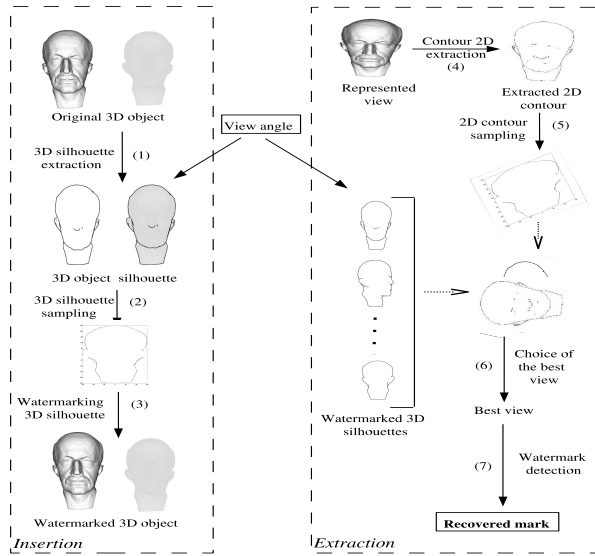


Fig. 2. Proposed framework.

mark it using a robust watermarking algorithm of 3D polygonal lines to get protected 3D object (step 3). This watermarked object can then be published for further use and representation in virtual scenes or hybrid natural/synthetic videos. Afterward, we can check if the represented object is protected by extracting its 2D contour (step 4) sampling it (step 5) and then detecting the watermark's presence (step 7).

As it can be seen, one view of a 3D object generally provides only a partial knowledge of the whole object. So it is better, if possible, to consider several view points to extract corresponding 3D silhouettes to be watermarked (e.g. Top view, side view and intermediate one of a human face).

As indicated in the figure 2, the knowledge of the projection matrix as well as intrinsic parameters of the virtual camera for watermark extraction are needed to overcome the basic problem of 2D/3D alignment [8], [9].

3. ALGORITHMIC DETAILS

This section is devoted to a detailed description of the most important steps of our approach.

3.1. Selection of anchor views

The choice of characteristic views has a great influence on the performance and limits of our watermarking approach. These viewing angles should satisfy the following practical constraints:

1. The represented views must cover all the 3D object.
2. Their number should be limited in order to avoid conflicting overlaps between parts of the watermarked sig-

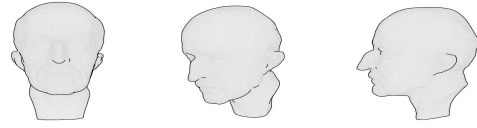


Fig. 3. Frontal view, intermediate view and side one.

nal, as well as to avoid excessive time computing.

For our first experiments we are interested in watermarking 3D faces. We chose k equal to 3 : frontal view, side view and intermediate one (see figure 3). We estimate (according to ongoing studies in the field of biometrics) that these three views are significant enough to characterize a face. In future work, we plan to use some existing techniques to provide an optimal selection of 2D views from a 3D model, in particular the one developed in [10] is under investigation. In this technique, the selection of characteristic views is based on adaptive clustering algorithm and using statistical model distribution scores to select the optimal number of views and their positions.

3.2. Contour Extraction

The usual method for computing 3D object silhouette is to iterate over every mesh edge and check:

1. If the current edge is associated to only one triangle, it is a contour.
2. If the current edge is associated to two triangles F_1 and F_2 ,
 - We note \vec{n}_1 and \vec{n}_2 normal vectors of F_1 and F_2 .
 - We note \vec{v} a vector composed by camera position and one vertex of the current edge.

If $(\vec{n}_1 \cdot \vec{v}) \cdot (\vec{n}_2 \cdot \vec{v}) < 0$ i.e. \vec{n}_1 and \vec{n}_2 have opposite direction w.r.t. the camera axis

Then F_1 and F_2 are oriented one to the front of the camera and the other one to its back therefore the current edge is a contour.

Else it is not an apparent contour edge.

3.3. Sampling 3D silhouettes

This step is required as we have to select vertices from the 3D silhouette to insert the mark. These same vertices must be reconsidered to extract the mark from the represented 2D view. To cope with this constraint, we propose the following method. First we compute the gravity center of the contour to be sampled, thereafter we consider lines $D_i : y = a_i x + b_i$ that pass through this gravity center. Both external intersections of the *ellipsoidal contour* described by the face with lines D_i are vertices to consider for the insertion and extraction of the mark. Figure 4 shows some results of 3D contour sampling.

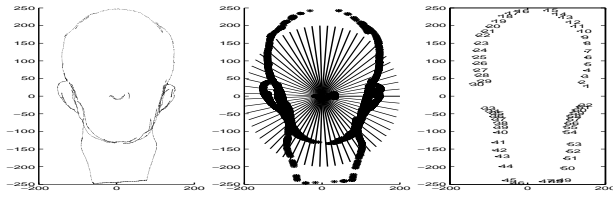


Fig. 4. Sampling 3D extracted silhouette.

It is worth mentioning that this process allows us to get a closed polygonal line from the 3-D and/or 2-D extracted contour.

3.4. Watermarking of 3D Silhouettes

To the best of our knowledge, there is no previous work dealing with watermarking of 3D polygonal lines. As far as we are concerned we have extended to 3D silhouettes the existing algorithm described in paper [11] and dealing with closed 2D contours. We first summarize the main steps of watermarking algorithm 2D contour and then we present our extension to closed 3D silhouettes.

3.4.1. Watermarking 2D contour

Let L_{2D} be a 2D contour that consists of N vertices, each of them represented as $[x(n), y(n)]$. Coordinates can be combined to construct the complex signal:

$$s_{2D}(n) = x(n) + i.y(n), \quad n = 0 : N. \quad (1)$$

Such a signal can be represented by its Fourier transform coefficients $S_{2D}(k)$, $k = 0 : N$.

Watermark construction and embedding: We note W_0 a bi-valuated ± 1 random sequence with zero mean and unit variance.

The watermark is constructed as follows:

$$W(i) = \begin{cases} 0 & i < aN \text{ or } bN < i < (1-b)N \\ & \text{or } (1-a)N < i. \\ W_0(i) & (1-b)N < i < (1-a)N \\ & \text{or } aN < i < bN. \end{cases}$$

The watermarked polygonal line is:

$$|S_{2D}^*(k)| = |S_{2D}(k)| + p|S_{2D}(k)| \cdot W(k). \quad (2)$$

a and b control the low and high frequency ranges that the watermark affects, $0 < a < b < 0.5$.

p determines the watermark strength and must be less than 1 to guarantee $|S_{2D}^*(k)|$ always positive.

The inverse Fourier transform of $S_{2D}^*(k)$ produces the watermarked polygonal line L_{2D}^* .

Watermark detection: Let $|S_{2D}^*(k)|$ be the Fourier descriptor of the watermarked line. The correlation coefficient c between W and $|S_{2D}^*(k)|$ informs us about the watermark's presence.

$$c = \sum (W(k) \cdot |S_{2D}^*(k)|) \quad (3)$$

Instead of c , a normalized correlation coefficient c' equal to $c/\text{mean}(c)$ is used. The detection is performed by comparing c' against a properly selected threshold T :

- H_0 : W watermarks L_{2D}^* if $c' > T$.
- H_1 : W does not watermark L_{2D}^* if $c' < T$.

3.4.2. Extension to 3D silhouettes

Let L_{3D} a 3D silhouette that consists of N vertices, each of them represented as $[x(n), y(n), z(n)]$. Coordinates can be combined as follows to construct the complex signal:

$$s_{3D}(n) = x(n)/z(n) + i.y(n)/z(n), \quad n = 0 : N. \quad (4)$$

To watermark this 3D silhouette we replace the signal s_{2D} defined in the equation 1 with s_{3D} (equation.4) and we use the same algorithm described below. We obtain watermarked coordinates $(x/z)^*$ and $(y/z)^*$. The watermarked 3D silhouettes L_{3D}^* is then defined by the N vertices:

$$[x^* = (x/z)^*.z, \quad y^* = (y/z)^*.z, \quad z]. \quad (5)$$

The choice of this complex signal s_{3D} to watermark 3D silhouettes L_{3D} is closely related to the main objective of our approach i.e. watermarking 3D object and retrieving information from represented views and by considering the basic equation linking 2D and 3D coordinates via a perspective projection which is defined as:

$$[x_p = f.x/z, y_p = f.y/z]. \quad (6)$$

where f is the focal length, $[x_p, y_p]$ are 2D coordinates after projection and $[x, y, z]$ are 3D coordinates.

3.4.3. Robustness to manipulation

Thanks to the adequate construction of the complex signal s_{3D} (equation.4) to insert/detect the mark, properties in terms of robustness against manipulations demonstrated in 2D remain valid in 3D.

- Translation of the represented view only affects the first Fourier descriptor $S_{3D}^*(0)$, by choosing $a > 0$ the watermark is robust to translation.
- Rotation by an angle θ of the represented view results in a multiplication by $\exp(i\theta)$ of the signal S_{3D}^* . The magnitude of the Fourier descriptors remain invariant. The algorithm is therefore robust to rotation.
- The normalization of the correlation coefficient c grants the robustness against scaling attacks.

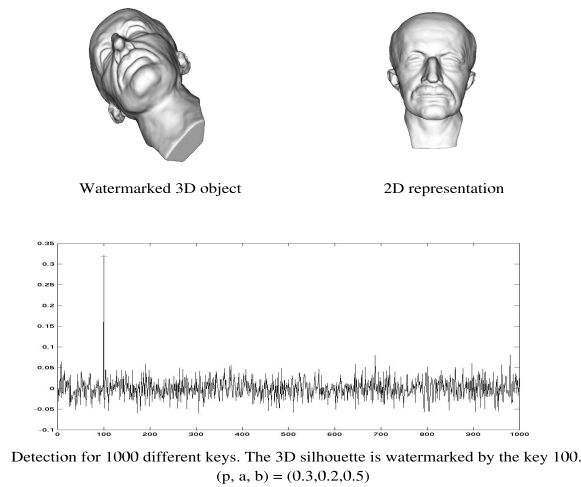


Fig. 5. Results obtained when watermarking a 3D object using its 3D silhouette and checking the watermark’s presence from a represented view.

4. RESULTS

We have carried out some preliminary experiments to provide a first assessment of the performances that can be expected from the presented 3D watermarking framework. Experiments we report here are conducted under minimal constraints and ideal conditions. We choose a single view angle to insert the mark in the corresponding 3D silhouette (top face of the model of a human head), the 3D object is projected in the 2D image regarding the same view angle as the one considered for the insertion with no alteration.

The human face used in the experiment is made up of 49132 vertices, the extracted 3D silhouette is composed of 5542 vertices after sampling. The obtained results are shown in figure 5.

5. CONCLUSION AND PERSPECTIVES

In this article we proposed an alternative framework for the watermarking of 3D objects. Whereas most of already published watermarking algorithms dealing with geometrical 3D objects aim at protecting the computer description of a 3D object regardless its visual representations, our research was driven by the expectation to protect the visual presentations of a 3D object in images or videos after it has been marked. We have also proposed an extension of 2D contour watermarking algorithm to a 3D silhouette.

As it can be seen, some steps of our approach are not fully designed and validated: experiments conducted in more realistic conditions (e.g. 2D represented view could coincide only partially with watermarked 3D silhouette), the automatic selection of characteristic views, the possible conflicting over-

laps between parts of the watermarked signal.

Mid term works will also concentrate on 3D object blind watermarking: the mark would be extracted from the 3D object’s represented views with no apriori knowledge about the set of parameters used to perform the projection.

6. REFERENCES

- [1] “French national rnr semantic 3d project, <https://www.semantic-3d.net/>”.
- [2] S. Ichikawa, H. Chiyama, and K. Akabane, “Redundancy in 3D polygon models and its application to digital signature,” *Journal of WSCG*, vol. 10, no. 1, pp. 225–232, 2002.
- [3] R. Ohbuchi, H. Masuda, and M. Aono, “Watermarking three-dimensional polygonal models,” in *ACM Multimedia*, Seattle, Washington, Nov. 1997, pp. 261–272.
- [4] T. Harte and A. G. Bors, “Watermarking 3D models,” in *IEEE International Conference on Image Processing*, Rochester, NY, USA, Sept. 2002.
- [5] O. Benedens, “Geometry-based watermarking of 3D models,” *IEEE Computer Graphics and Applications*, vol. 19, no. 1, pp. 46–55, 1999.
- [6] R. Ohbuchi, H. Masuda, and M. Aono, “A shape-preserving data embedding algorithm for NURBS curves and surfaces,” in *Computer Graphics International*, June 1999, pp. 170–177.
- [7] E. Garcia and J.-L. Dugelay, “Texture-based watermarking of 3D video objects,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, Aug. 2003.
- [8] R. Basri and W. Jacobs, “Projective alignment with regions,” in *IEEE Transactions on pattern analysis and machine intelligence*, 2001, vol. 23, pp. 519–527.
- [9] J. Feldmar, N. Ayache, and F. Betting, “3d-2d projective registration of free-form curves and surfaces,” in *Computer Vision and Image Understanding*, 1997, vol. 65, pp. 403–424.
- [10] T. Filali, M. Daoudi, and J-P. Vandeborre, “3d model retrieval based on adaptive views clustering,” in *3rd International Conference on Advances in Pattern Recognition (ICAPR)*, 2005.
- [11] Vassilios Solachidis and Ioannis Pitas, “Watermarking polygonal lines using fourier descriptors,” *IEEE Computer Graphics and Applications*, vol. 24, no. 3, pp. 44–51, 2004.