A BLIND ROBUST WATERMARKING SCHEME FOR COPYRIGHT PROTECTION OF 3D MESH MODELS

Stefanos Zafeiriou , Anastasios Tefas and Ioannis Pitas

Dept. of Informatics, Aristotle University of Thessaloniki, Box 451, 54124 Thessaloniki, Greece
e-mail: {dralbert,tefas,pitas}@zeus.csd.auth.gr

ABSTRACT

In this paper, a novel method for blind 3D mesh model watermarking applications is proposed. The method is robust against 3D translation, scaling and mesh simplifications. A pseudo-random watermarking signal is casted in the 3D mesh model by deforming geometrically its vertices, without altering the vertex topology. Prior to embedding and detection a set of simple transforms is applied to the 3D mesh model. Each sample of the watermark sequence is embedded in a set of vertices rather than in a single vertex in order to deal with mesh simplifications. Experimental results indicate the ability of the proposed method to deal with the aforementioned attacks.

1. INTRODUCTION

Watermarking ages about a decade as a mean of copyright protection and content verification. Even though watermarking is a very active research field and it's application to 2D still images and audio signals has been thoroughly studied, watermarking of 3D mesh models, has not attract much the researchers' attention.

Digital watermarking of 3D models remains a difficult problem due to the fact that there is no unique representation of 3D models, e.g. 3D mesh based models, 3D models represented using parametric surfaces such as 3D NURBS graphic data [1] (i.e, 3D data that are represented using Nonuniform Rational B-Spline surfaces) or 3D data combined with texture information [2].

In copyright protection watermarking systems, a digital watermark is embedded in the host signal using the owners private key and can be detected only using the same key. The embedded watermark should be perceptually invisible and statistically undetectable.

In [3], Benedens described a watermarking system that is based on affine registration of meshes in order to compensate for affine transformations and used it in the watermarking detection procedure. The main limitation of the previous method, which is a major drawback for a watermarking technique, is that it is not blind due the registration method that requires both the original and the test 3D mesh model.

Song et al. proposed a robust watermarking algorithm that transforms the mesh into an image and then embeds the watermark using image-based watermarking techniques [4]. The algorithm is robust against translation, rotation, scaling, mesh simplification and gaussian noise attack but it requires additional information for watermark detection and thus, it is not blind.

Harte and Bors proposed a relative simple algorithm for watermarking 3D mesh model using blind detection [5]. They used a technique for embedding the watermark in vertices that fulfill a certain geometric criterion. The authors claim that this criterion ensures very low perceptibility of the watermark. The watermark is robust against rotation, translation, uniform scaling and cropping. However results against more sophisticated attacks such as mesh simplification have not been published.

This paper focuses on watermarking of 3D models, that are represented using mesh information (a list of vertices) and an arbitrary connectivity (a list of connections between vertices), for copyright protection applications and is an extension of the [6] in order to handle mesh simplification attacks. This is the first blind 3D watermarking method that copes with mesh simplification and geometric attacks according to the authors knowledge of the watermarking literature.

2. 3D MESH MODEL TRANSFORM

A 3D mesh model is comprised of a set of vertices $\mathbf{V}^c$ in cartesian coordinates and a set of connections between these vertices. Let $\mathbf{v}_i^c$ be the i vertex which is comprised of three coordinates in cartesian space, $\mathbf{v}_i^c = (x_i, y_i, z_i)$. The representation of the vertex $\mathbf{v}_i^c$ in spherical coordinates is $\mathbf{v}_i^s = (r_i, \theta_i, \phi_i)$. The set of all vertices of the 3D mesh model in spherical coordinates will be denoted as $\mathbf{V}^s$. For the cardinality of a set $\mathbf{X}$ the notation $N(\mathbf{X})$ will be used.

The first step of the watermarking procedure is a trans-

The work presented was developed within VISNET, a European Network of Excellence (http://www.visnet-noe.org), funded under the European Commission IST FP6 programme.

form of the 3D mesh model prior to watermark embedding as well as prior to watermark detection. The objective of the 3D mesh model transform is to obtain invariance against translation and rotation. A description of each step of the transform follows.

- **Model Translation**. The model is translated so that the center of mass falls on the center of the axes. To find the center of mass the following equation is used

$$\mathbf{K} = \frac{1}{N(\mathbf{V}^c)} \sum_i \mathbf{v}_i^c \tag{1}$$

  where $\mathbf{v}_i^c$ is the i-th vertex .

- **Model Rotation**. The model is rotated so that its principal component $\mathbf{u}$ coincides with the z axis. The principal component $\mathbf{u}$ of the vertices is the eigenvector that corresponds to the greatest eigenvalue of the covariance matrix $\mathbf{C}$ of the vertices coordinates. The covariance matrix $\mathbf{C}$ is calculated as follows:

$$\mathbf{C} = \begin{bmatrix} \sum_{i=0}^{N(\mathbf{V}^c)} x_i^2 & \sum_{i=0}^{N(\mathbf{V}^c)} x_i y_i & \sum_{i=0}^{N(\mathbf{V}^c)} x_i z_i \\ \sum_{i=0}^{N(\mathbf{V}^c)} x_i y_i & \sum_{i=0}^{N(\mathbf{V}^c)} y_i^2 & \sum_{i=0}^{N(\mathbf{V}^c)} y_i z_i \\ \sum_{i=0}^{N(\mathbf{V}^c)} x_i z_i & \sum_{i=0}^{N(\mathbf{V}^c)} z_i y_i & \sum_{i=0}^{N(\mathbf{V}^c)} z_i^2 \end{bmatrix} \tag{2}$$

  where x_i, y_i and z_i are the coordinates of the vertex $\mathbf{v}_i^c$. Thus, robustness against rotation of the watermarked model is achieved.

- **Conversion to Spherical Coordinates**. The model is converted to spherical coordinates in order to achieve robustness against scaling. To do so, the watermark is embedded in the r component of each vertex.

The notations $r(\mathbf{v}_i^s)$, $\theta(\mathbf{v}_i^s)$, $\phi(\mathbf{v}_i^s)$ will be used for the r_i, θ_i, ϕ_i components of the vertex $\mathbf{v}_i^s$. For each 3D model a set $\mathbf{\Theta}$ is defined as $\mathbf{\Theta} = \{\theta_j : \exists \mathbf{v}_i^s \in \mathbf{V}^s, \theta(\mathbf{v}_i^s) = \theta_j\}$.

3. WATERMARKING PROCEDURE

In order to achieve robustness against mesh simplification every watermark sample is embedded to a set of vertices instead of one vertex. Thus, a set of vertices that correspond to a range of θ angles $\mathbf{\Theta}_j \subset \mathbf{\Theta}$ is selected and the r components of these vertices are used as embedding primitive.

Let the set $\mathbf{I}(\mathbf{\Theta}_j) = \{\mathbf{u}_i^s : \theta(\mathbf{u}_i^s) \in \mathbf{\Theta}_j\}$ and for this set the random variable $d_r(\mathbf{u}_i^s)$ is formed as:

$$d_r(\mathbf{u}_i^s) = r(\mathbf{u}_i^s) - H(\mathbf{u}_i^s) \tag{3}$$

where H is a local neighborhood operation of the vertices around $\mathbf{u}_i^s$ and $H(\mathbf{u}_i^s)$ is an approximation function of $r(\mathbf{u}_i^s)$ that depends on the neighborhood of $\mathbf{u}_i^s$. The operator H is chosen so that $d_r(\mathbf{u}_i^s)$ follows a Gaussian distribution with variance σ^2 and zero mean. The operator H should also have the property:

$$H(\mathbf{u}_i^s) = aH(\mathbf{v}_i^s) \tag{4}$$

where $\mathbf{u}_i^c = a\mathbf{v}_i^c$ in the corresponding cartesian coordinates and a is a scalar that corresponds to the scaling factor. The former property grands the methods robustness against uniform scaling.

3.1. Watermark Generation

The watermark generation in this scheme aims at separating the interval $[0, \pi]$ in W^θ intervals $\mathbf{\Theta}_j$. Each interval $\mathbf{\Theta}_j$ is used for embedding a label $l(\mathbf{\Theta}_j) \in \{-1, 0, 1\}$. The value $l(\mathbf{\Theta}_j)$ for each interval, is determined by the owners key K using a pseudo-random number generator. The intervals $\mathbf{\Theta}_j$ for which $l(\mathbf{\Theta}_j) \in \{-1, 1\}$ have fixed length of t rad. The length t is determined by the tolerance that the algorithm should have in case of alterations on the principal component.

3.2. Watermark Embedding

The watermark is embedded in the 3D mesh model after the application of the transforms described in Section 2, by altering the r component of the vertices of $\mathbf{I}(\mathbf{\Theta}_j)$ according to:

$$\mathbf{I}^w(\mathbf{\Theta}_j) = \begin{cases} \mathbf{I}(\mathbf{\Theta}_j) & \text{if } l(\mathbf{\Theta}_j) = 0 \\ \mathbf{G}_1(\mathbf{I}(\mathbf{\Theta}_j)) & \text{if } l(\mathbf{\Theta}_j) = 1 \\ \mathbf{G}_2(\mathbf{I}(\mathbf{\Theta}_j)) & \text{if } l(\mathbf{\Theta}_j) = -1 \end{cases} \tag{5}$$

where $\mathbf{G}_1$ and $\mathbf{G}_2$ when applied to a set $\mathbf{I}(\mathbf{\Theta}_j)$, cast a watermark sample $l(\mathbf{\Theta}_j)$ by changing the distribution of the random variable d_r of the vertices contained in $\mathbf{I}(\mathbf{\Theta}_j)$. In order to detect these changes the variance σ^2 of the d_r must remain almost intact after the embedding. Due to the fact that d_r follows Gaussian distribution with variance σ^2 and zero mean the left σ_l^2 and right σ_r^2 variance estimators are sufficient for evaluating σ^2. The estimators σ_l^2 and σ_r^2 for a random variable x having zero mean are defined as:

$$\sigma_l^2 = \frac{1}{N(\{x : x < 0\}) - 1} \sum_{x<0} x^2 \tag{6}$$

$$\sigma_r^2 = \frac{1}{N(\{x : x > 0\}) - 1} \sum_{x>0} x^2 \tag{7}$$

and

$$\sigma_l^2 \approx \sigma_r^2 \approx \sigma^2. \tag{8}$$

In the embedding procedure $\mathbf{G}_1$ changes the distribution of d_r by inducing deformations in the r component of the

vertices of a set $\mathbf{I}(\mathbf{\Theta}_j)$ without altering the σ_l^2 of d_r. The application of $\mathbf{G}_1$ alters the r component of some of the vertices $\mathbf{v}_i^s$ that have $d_r(\mathbf{v}_i^s) > b\sigma_l$ in order to fall inside $(0, b\sigma_l)$. In the same manner $\mathbf{G}_2$ deforms the distribution of d_r by altering the r component of some of the vertices $\mathbf{v}_i^s$ that have $d_r(\mathbf{v}_i^s) < -b\sigma_r$ in order to fall inside $(-b\sigma_r, 0)$ without altering the σ_r^2 of d_r.

That is, watermark embedding is performed by altering only some of the vertices $\mathbf{v}_i^s$ with $d_r(\mathbf{v}_i^s) > 0$ when embedding the watermark sample 1, ($l(\mathbf{\Theta}_j) = 1$) whereas the remaining vertices $\mathbf{v}_i^s$ for which $d_r(\mathbf{v}_i^s) < 0$ stay unaltered since they are used in the detection procedure. If $l(\mathbf{\Theta}_j) = -1$ the vertices with $d_r(\mathbf{v}_i^s) > 0$ remain unaltered and those with $d_r(\mathbf{v}_i^s) < 0$ are used for embedding. The constant b controls the watermark perceptibility.

For an unwatermarked 3D mesh model for a set of vertices $\mathbf{I}(\mathbf{\Theta}_j)$ of a 3D mesh model it is valid that:

$$Prob(d_r > b\sigma_l) = \frac{N(\mathbf{I}_r(\Theta_j))}{N(\mathbf{I}(\Theta_j))} \approx G(-b) \tag{9}$$

and

$$Prob(d_r < -b\sigma_r) = \frac{N(\mathbf{I}_l(\Theta_j))}{N(\mathbf{I}(\Theta_j))} \approx G(-b) \tag{10}$$

where $\mathbf{I}_r(\mathbf{\Theta}_j) = \{\mathbf{u}^s \in \mathbf{I}(\mathbf{\Theta}_j) : d_r(\mathbf{u}^s) > b\sigma_l\}$ and $\mathbf{I}_l(\Theta_j) = \{\mathbf{u}^s \in \mathbf{I}(\mathbf{\Theta}_j) : d_r(\mathbf{u}^s) < -b\sigma_r\}$. The function G is given by:

$$G(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{\frac{-y^2}{2}} dy \tag{11}$$

and it is known that $G(-x) = 1 - G(x)$.

Let $\mathbf{I}_r^w(\mathbf{\Theta}_j)$ and $\mathbf{I}_l^w(\mathbf{\Theta}_j)$ (in the watermarked 3D mesh model) be the corresponding sets of $\mathbf{I}_r(\mathbf{\Theta}_j)$ and $\mathbf{I}_l(\mathbf{\Theta}_j)$ of the original 3D mesh model.

For the watermarked 3D mesh model and for the set $\mathbf{I}^w(\mathbf{\Theta}_j)$ that has been produced by $\mathbf{G}_1$ the following inequality is valid:

$$Prob(d_r > b\sigma_l) = \frac{N(\mathbf{I}_r^w(\mathbf{\Theta}_j))}{N(\mathbf{I}^w(\mathbf{\Theta}_j))} < G(-b) \tag{12}$$

Similarly if $\mathbf{I}^w(\mathbf{\Theta}_j)$ was created by $\mathbf{G}_2$ the corresponding inequality is valid:

$$Prob(dr < -b\sigma_l) = \frac{N(\mathbf{I}_l^w(\mathbf{\Theta}_j))}{N(\mathbf{I}^w(\mathbf{\Theta}_j))} < G(-b) \tag{13}$$

Equations (9), (10), (12) and (13) are used for calculating the detection ratio which is used for deciding whether a model is watermarked or not.

The watermark embedding procedure is an iterative procedure applied to each interval $\mathbf{\Theta}_j$ and ends when the interval $[0, \pi]$ is covered. In the first step a number $\theta^k(1) \in [0, \pi]$ is picked from a pseudo-random number generator using the owners private key. Afterwards, at each step m of the procedure two uniform distributed pseudo-random number generators are used for producing a number $w(m) \in \{-1, 1\}$ and a random angle $\theta_1(m) \in (0, \epsilon)$. The value $w(m)$ is used for labelling the set $\mathbf{I}([\theta^k(m), \theta^k(m) + t))$ and the set $\mathbf{I}([\theta^k(m) + t, \theta^k(m) + t + \theta_1(m)))$ remains unaltered (labelled with 0). Equation (5) is used for embedding the watermark in these sets and $\theta^k(m+1)$ is set equal to $\theta^k(m) + t + \theta_1(m)$. The algorithm continues in the same way until the interval $[0, \pi]$ is covered.

3.3. Watermark Detection

Prior to watermark detection the 3D mesh model under investigation is transformed as described in Section 2. Afterwards, the watermark sequence is generated according to the owners key K forming the intervals $\mathbf{\Theta}_j$ and the labels $l(\mathbf{\Theta}_j)$. For the sets $\mathbf{I}^w(\mathbf{\Theta}_j)$ with $l(\mathbf{\Theta}_j) \in \{-1, 1\}$ the detection signal is formed as :

$$d(\mathbf{\Theta}_j) = \begin{cases} \frac{N(\mathbf{I}_r^w(\mathbf{\Theta}_j))}{N(\mathbf{I}^w(\mathbf{\Theta}_j))} & \text{, if } l(\mathbf{\Theta}_j) = 1 \\ \frac{N(\mathbf{I}_l^w(\mathbf{\Theta}_j))}{N(\mathbf{I}^w(\mathbf{\Theta}_j))} & \text{, if } l(\mathbf{\Theta}_j) = -1 \end{cases} \tag{14}$$

The detection ratio is given by the mean value of the signal $d(\mathbf{\Theta}_j)$ of the 3D mesh model as:

$$D_w = \frac{1}{N(\mathbf{M})} \sum_{j \in \mathbf{M}} d(\mathbf{\Theta}_j) \tag{15}$$

where $\mathbf{M} = \{k : l(\mathbf{\Theta}_k) \in \{-1, 1\}\}$. For an unwatermarked 3D mesh model:

$$D_w \approx G(-b) \tag{16}$$

whereas for a watermarked model :

$$D_w < G(-b). \tag{17}$$

The decision about the ownership of the 3D mesh model is taken by comparing the watermark detection ratio of the model to a predefined threshold T.

4. EXPERIMENTAL RESULTS

A set of experiments using several 3D mesh models has been conducted to illustrate the robustness of the proposed technique against several geometric attacks and 3D mesh simplification.

The geometric attacks that were considered are translation, rotation and uniform scaling. Due to the invariance properties of the transform that is applied to the model prior to watermark embedding and detection, the results for these attacks were identical to the ones obtained when no attack is

considered. Thus, they will not be presented separately and only experimental results against mesh simplification will be given due to lack of space.

From the experiments it was found that the principal component of the simplified model differs from the original model's principal component about 0.7 degrees for mesh simplification up to 40%. Thus, the parameter t that controls the length of the interval $\mathbf{\Theta}_j$ was chosen to be 1.4 for achieving robustness against mesh simplification. The parameter b that controls the watermark perceptibility was set to 0.6 and thus, $G(-b) = 0.2742$.

The 3D model used for demonstrating the results is the Foot with 25845 vertices and 51690 triangles. The original model can be seen in Figure 1 whereas the watermarked model with $t = 1.4$ and $b = 0.6$ is depicted in Figure 2. The model has been watermarked using 1000 random keys and then simplified with various mesh simplification factors that range from 15% to 50%. Detection has been performed using the 1000 correct and 1000 wrong keys. The corresponding ROC curves are depicted in Figure 3.

Fig. 1. Foot original model.

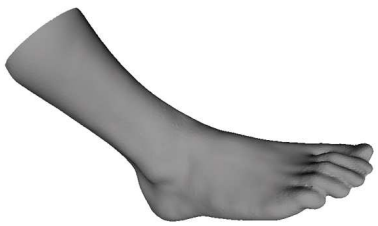

Fig. 2. Foot watermarked model, rotated and scaled.

The Equal Error Rate (EER) for the Foot without mesh simplification was found to be about 10^{-20} and is reduced to 10^{-3} after mesh simplification up to 50% using the simplification algorithm reported at [7].

5. CONCLUSIONS

A novel blind 3D model watermarking method has been proposed in this paper. The proposed algorithm is robust against geometric attacks such as translation, rotation and uniform scaling. It is also robust against mesh simplification without using extra information during the detection procedure. Experimental results presented in this paper indicate the appropriateness of the proposed algorithm in copyright protection of 3D models.

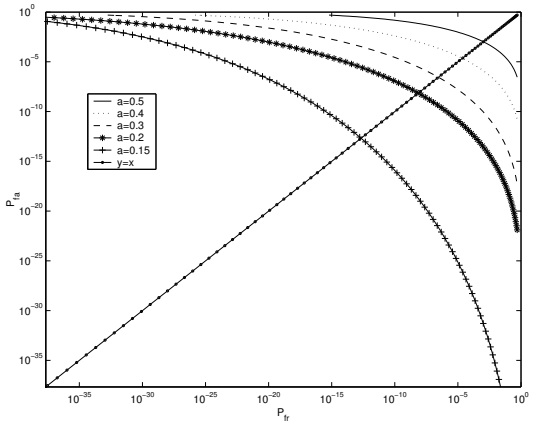

Fig. 3. ROC curves of Foot model for various vertex decimation percentages (a is the simplification rate).

6. REFERENCES

[1] J. Lee, N. Cho, and J. Nam, "Watermarking for 3d nurbs graphic data," in *IEEE International Workshop on MMSP 2002*, December 2002, pp. 304–307.

[2] F. Garcia and J. Dugelay, "Texture-based watermarking of 3-d video objects," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 853–866, August 2003.

[3] O. Benedens, "Robust watermarking and affine registration of 3d meshes," *Information Hiding, no. 2578 in Lecture Notes in Computer Science*, pp. 177–195, 2003.

[4] H. Song, N. Cho, and J. Kim, "Robust watermarking of 3d mesh models," in *IEEE International Workshop on MMSP 2002*, December 2002, pp. 332–335.

[5] T. Harte and A.G. Bors, "Watermarking 3-d models," in *Proceedings IEEE International Conference on Image Processing*, Rochester, NY, USA, Semptember 22-25 2002, vol. III, pp. 661–664.

[6] A.Kalivas, A.Tefas, and I.Pitas, "Watermarking of 3d models using principal component analysis," in *Proceedinds of IEEE International Conference on Acoustics, Speech, and Signal Processing*, April 2003, vol. V, pp. 676–679.

[7] W.J. Shroder, J.A. Zarge, and W.E. Lorensen, "Decimation of triangle meshes," in *SIGGRAPH 92 Proceedings*, 1992, pp. 65–70.