3D MESH WATERMARKING USING PROJECTION ONTO CONVEX SETS

Suk-Hwan Lee, Tae-Su Kim,* Seung-Jin Kim,* Young Huh,** Ki-Ryong Kwon,*** Kuhn-Il Lee**

*School of Electrical Engineering and Computer Science, Kyungpook National University
1370 Sangyuk-Dong, Buk-Gu, Daegu 702-701, Republic of Korea
** Korea Electrotechnology Research Institute
***Department of Electronic Engineering, Pusan University of Foreign Studies
55-1 Uam-Dong, Nam-Gu, Pusan 608-738, Republic of Korea

ABSTRACT

The current paper proposes 3D mesh watermarking using projection onto convex sets. As such, a 3D mesh is iteratively projected onto two constraint convex sets until the convergence condition is satis£ed. The sets consist of a robustness set and invisibility set designed to embed the watermark. The watermark can be extracted without the original mesh using the decision values and index that the watermark was embedded with. Experimental results verify that the watermarked mesh is both robust to mesh simpli£cation, cropping, af£ne transformations, and vertex randomization and invisible.

1. INTRODUCTION

There has been a recent increased interest in Web 3D techniques that realize 3D graphics. Among such techniques, the international standards organization, ISO/IEC, has con£rmed VRML as the standard for representing 3D graphics on the Web, which has made sources of VRML available to the general public. Thus, 3D watermarking has recently been a focus of research to protect the copyright of VRML. VRML can be represented as a mesh de£ned by vertices and connectivity. However, this mesh has different characteristics to a still image, video, or audio, as it can be easily edited by a geometrical and topological operator, and represented by many descriptors. Plus, the order of arrangement of the vertices is not de£nitive. Also, to increase the rendering speed, the data for the vertices is reduced by mesh simpli£cation while preserving the original shape. Therefore, 3D watermarking must consider all these characteristics.

Ohbuchi[3] proposed an algorithm that adds a watermark to the spectral domain of a 3D polygonal mesh by projecting the coordinates of the vertices onto a set of eigenvectors. Praun[4] proposed a generalizing spread spectrum technique for an arbitrary mesh derived from progressive meshes to construct a multi-resolution set of scalar basis functions over the mesh. Both of the above algorithms require a resampling process using the original mesh connectivity to extract the watermark from a mesh that has been attacked by modifying the mesh connectivity. Therefore, Benedens[5] proposed a watermarking method that modi£es the model's normal distribution to store information solely in the geometry of the model. As such, this algorithm is robust to randomization of the vertices, re-meshing, and mesh simpli£cation. Yet, if the watermarked model is attacked by partial resection, such as cropping, the watermark embedded in that section will disappear, whereas for an af£ne transformation, this algorithm needs to be realigned with the original model.

Accordingly, the current paper proposes blind watermarking based on projection onto convex sets that is robust to topological and geometric attacks, and imperceptible. A binary watermark is then embedded by modifying the sample means of the r coordinate values sampled into bins with a high vertex density. To modify the sample means while achieving robustness and invisibility, the mesh model is projected onto two constraint convex sets that satisfy the condition of robustness and invisibility. The watermark can be extracted without the original model using the index information of the embedded bins and decision values for each bin. Experimental results verify that the proposed algorithm is both robust to af£ne transformations, mesh simpli£cation, cropping, and vertex randomization and imperceptible.

2. PROJECTION ONTO CONVEX SETS

The theory of projection onto convex sets (POCS) has already been applied to various image recovery problems.[1] Assume that all images, represented as $N^2 \times 1$ vectors, are elements of a Hilbert space $\mathbf{H}$. Given m closed convex sets $C_i, (i = 1, 2, \cdots, m)$ in $\mathbf{H}$, the iteration is $\mathbf{f}_{n+1} = P_m P_{m-1} \cdots, P_1 \mathbf{f}_n$, $n = 1, 2, \cdots$, where the projection of P_i onto C_i is de£ned by $\| \mathbf{f} - P_i \mathbf{H} \| = \min_{\mathbf{g} \in C_i} \| \mathbf{f} - \mathbf{g} \|$, where $\mathbf{g}$ is the projection of $\mathbf{f}$, which will converge to the point $C_i = \bigcap_{i=1}^{m} C_i$ nonempty sets for any initial $\mathbf{g}_0$. The key idea in applying the theory of POCS to recovery problems is to represent every known property of the original by a closed convex set. Therefore, for m known properties, there are m closed convex sets C_i. Then, the vector $\mathbf{f}^*$, common to all sets $C_i, (i = 1, 2, \cdots, m)$, can be found by alternating projections onto each one of them, starting from any initial guess vector. Clearly, the point of convergence $\mathbf{f}^*$ then possesses all the m desired properties of the original. As such, to apply the theory of POCS to 3D mesh watermarking, for a 3D mesh $\mathbf{M}$ with $N \times 1$ vectors (N vertices) in $\mathbf{H}$, a vertex $\mathbf{v}$ of one vector is projected into the constraint sets, thereby satisfying the pursuit of watermark embedding.

3. THE PROPOSED 3D MESH WATERMARKING

3.1. The watermark embedding scheme

Given 3D mesh $\mathbf{M}$ with N vertices $\mathbf{v}$ in $\mathbf{R}^3$, it is translated so that the center of mass falls on the origin of axes, $\mathbf{v}^T = \mathbf{v} - 1/N \sum_{i=1}^{N} \mathbf{v}_i$. The vertices of the translated model $\mathbf{M}^T$ are converted to the spherical coordinates (r, θ, φ), and all components of $r_j (j = 1, 2, \cdots, N)$ are sampled into K bins with the uniform distance as follows;

$$Q(r_j) = \mathrm{INT}(\frac{r_j - r_{\text{min}}}{r_{\text{max}} - r_{\text{min}}}) \times K + 0.5 \qquad (1)$$

K is determined to consider the bit number for watermark embedding. $r_{\text{max}}, r_{\text{min}}$ are respectively max. and min. value of r component. Sample means, $E_k[r]$ of $r = Q^{-1}(k)$ in each of the sampled bins are calculated and arranged as the descending order according to the density of vertex per bin. We embed the binary watermark $w_k \in \mathbf{W}$ with $N_w \leq K$ length into the sample means as the descending order as $E'_k[r] = (1 + \alpha R_k) \times E_k[r]$. If $w_k = 0$, then R_k is -1. Otherwise, R_k is 1. α is the embedding strength. To extract the watermark without the original mesh, the watermark keys are required. They are the indices of bins into which the watermark is embedded and sample mean $E_k[r]$, written by E_k in this paper, before the watermark embedding. In the next subsection, $E_k[r]$ is the sample mean calculated at each iteration for projection onto convex sets. Actually the sample mean in any bin is not the center value of its bin. So, the partition bin size is reassigned so that E_k is the center value of bin and it used to the decision value for the watermark detection.

3.2. Constraint sets and projectors

For the above scheme of watermark embedding, we construct the two geometric constraint sets and their projectors such as the robustness constraint and the visibility constraint. The former is global constraint set and the latter is the local constraint set.

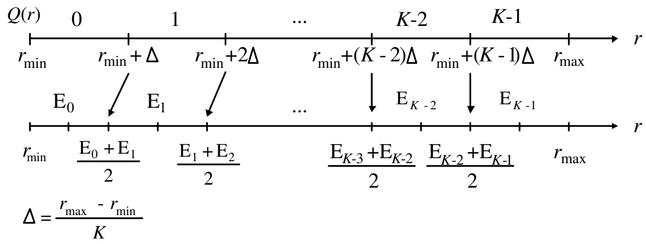

Fig. 1. All r components are sampled from K bins with a uniform distance. The sample means E from each bin are used to extract the watermark.

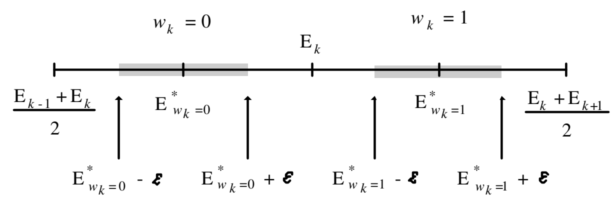

Fig. 2. Location into which the watermark is embedded for the k th bin.

3.2.1. *The robustness constraint set*

The sample mean $E'_k[r]$ that watermark embedded has to be at the available location to have the robustness. If $w_k = 1$, then $\mathrm{E}_k \leq E'_k[r] < (\mathrm{E}_k + \mathrm{E}_{k+1})/2$. $E'_k[r]$ has to be near to the center point between max and min. The closed convex set of the robustness constraint can be de£ned as

$$C_r = \{\mathbf{M}^T |\, |E_k[r] - E^*_{w_k}| \leq \varepsilon, k = 1, 2, \cdots, N_w\} \qquad (2)$$

where $\mathrm{E}^*_{w_k} = \begin{cases} (3\mathrm{E}_k + \mathrm{E}_{k+1})/4, & \text{if } w_k = 1 \\ (3\mathrm{E}_k + \mathrm{E}_{k-1})/4, & \text{otherwise} \end{cases}$, and $\varepsilon = \begin{cases} ((\mathrm{E}_{k+1} + \mathrm{E}_k)/2 + \mathrm{E}^*_{w_k})/2 - \mathrm{E}^*_{w_k})/c, & \text{if } w_k = 1 \\ (\mathrm{E}^*_{w_k} - ((\mathrm{E}_{k-1} + \mathrm{E}_k)/2 + \mathrm{E}^*_{w_k})/2)/c, & \text{otherwise} \end{cases}$. This set can be easy to be closed and convex. The geometric projection $\mathbf{M'} = P_r\mathbf{M}^T$ onto C_r can be de£ned as follows. If $w_k = 1$, then

$$E'_k[r] = \alpha(\mathrm{E}^*_{w_k=1} + \varepsilon) + (1 - \alpha)E_k[r] \qquad (3)$$

where $\frac{(\mathrm{E}^*_{w_k=1} - \varepsilon) - E_k[r]}{(\mathrm{E}^*_{w_k=1} + \varepsilon) - E_k[r]} \leq \alpha \leq 1$. And if $w_k = 0$, then

$$E'_k[r] = \alpha(\mathrm{E}^*_{w_k=0} - \varepsilon) + (1 - \alpha)E_k[r] \qquad (4)$$

where $\frac{E_k[r] - (\mathrm{E}^*_{w_k=0} + \varepsilon)}{E_k[r] - (\mathrm{E}^*_{w_k=0} - \varepsilon)} \leq \alpha \leq 1$. α is determined to the intermediate value between max and min.

3.2.2. *The invisibility constraint set*

The invisibility constraint set C_v limits the change of r component and is de£ned as

$$C_v = \{\mathbf{M}^T | \, R_i^L \leq r_i \leq R_i^H, i = 1, 2, \cdots, N\} \quad (5)$$

where R_i^L, R_i^H are respectively lower and upper bound for r_i component of i th. vertex. It is easily shown that this set is also closed and convex. A vertex $\mathbf{v}_i$ must be changed within the range that is below each Cartesian coordinate value of valence vertices $\mathbf{v}_i = (x_{ia}, y_{ia}, z_{ia})$ connected to it. The available range for a vertex is $x_i = [x_i - \triangle x, x_i + \triangle x]$, $y_i = [y_i - \triangle y, y_i + \triangle y]$, $z_i = [z_i - \triangle z, z_i + \triangle z]$ where $\triangle x = \min|mx_i - x_{ia}|/2$, $\triangle y = \min|my_i - y_{ia}|/2$, $\triangle z = \min|mz_i - z_{ia}|/2$, and $mx_i = \sum_{a=1}^{n_i} x_{ia}$, $my_i = \sum_{a=1}^{n_i} y_{ia}$, $mz_i = \sum_{a=1}^{n_i} z_{ia}$. n_i is the number of valence vertex. This projector P_v onto the set C_v, which takes the limitation of the each $r_i \in \mathbf{R}, i = 1, 2, \cdots, N$, considering the environment of its vertex $\mathbf{v}_i$, is followed by

$$r_i = \begin{cases} R_i^H, & \text{if } r_i > R_i^H \\ R_i^L, & \text{else if } r_i < R_i^L \\ r_i, & \text{else} \end{cases} \quad (6)$$

where $R_i^H = r_{i,avg} + \delta$, $R_i^L = r_{i,avg} - \delta$, $\delta = \sqrt{\triangle x^2 + \triangle y^2 + \triangle z^2}$, and $r_{i,avg} = \sqrt{mx_i^2 + my_i^2 + mz_i^2}$.

3.2.3. *POCS-based watermarking*

Using the previously de£ned sets C_r and C_v, the resulting 3D mesh embedded watermark can be described by the following steps; 1) Take the original model as the initial model $\mathbf{M}_0$. 2) Compute $\mathbf{M}_n = P_v P_r \mathbf{M}_{n-1}$ for each iteration $n = 1, 2, \cdots$. 3) Iterate continuously until $\sum_{k=0}^{N_w} \| E'_{k,n} - E'_{k,n-1} \|^2 \simeq 0$, where $E'_{k,n}$ and $E'_{k,n-1}$ are respectively sample means in k th. bin of $\mathbf{M}_n$ and $\mathbf{M}_{n-1}$. Sample means of $\mathbf{M}_0$ used as the decision value $r_{\text{max}}, r_{\text{min}}$ and the index information into which watermark is embedded are stored to extract watermark and it's the descending order of sample bins according to the vertex density.

3.3. Watermark extraction

Before extracting a watermark from an attacked model, for example after an af£ne transformation, realignment is necessary. When the watermarked model is only attacked by an af£ne transformation, such as scaling and translation, the attacked model still preserves the topology of the original model. Thus, the attacked model can easily be re-scaled to have the same dynamic range $r_{\text{max}} - r_{\text{min}}$ of r component in the original model and translated to make the center mass the origin. However, if the watermarked model is attacked by an af£ne transformation with topology deformation, such as mesh simpli£cation or cropping, it is dif£cult to realign this attacked model without the original model. The origin of the attacked model is found using the information stored for extracting the watermark. First, the vertex density of each bin into which the component from the center mass has been sampled is calculated, then the bins are arranged in a descending order according to their density. If this order information is not the same as the stored order information, the reference origin is searched for until the two orders are the same. After realignment, the process of watermark extraction is similar to that of watermark embedding. The sample means of the attacked model are calculated and the sample mean $\tilde{\text{E}}_k$ of the bin into which the watermark has been embedded is compared to E_k, the stored decision value. Thus, $w_k = 1$, if $\text{E}_k - \tilde{\text{E}}_k$ and otherwise, $w_k = 0$.

4. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed algorithm, computer simulations were performed using VRML data for the Stanford Bunny and Knots.[7] The binary watermark was a 50 length Gaussian random sequence converted to 1 bit. The number of sample bins K was 100 and the watermark was embedded into the 50 bins with the highest vertex density. The meshes with the watermark embedded using the proposed algorithm are shown in £gure 3. No objective evaluation for visibility, such as the PSNR in image processing, has yet been adopted for 3D graphics. Therefore, a subjective evaluation was used to verify that the watermark was imperceptible. To evaluate the robustness, the watermark-embedded models were attacked by mesh simpli£cation, rotation through an af£ne transformation, cropping, and vertex randomization. The strength of the attacks was also adjusted, such as the % in the mesh simpli£cation and a in the vertex randomization, to produce bit error. The robustness against these attacks is shown in Table I, which uses the bit error rate, BER. The watermark-embedded model was attacked with mesh simpli£cation using MeshToSS.[7] The % in the table represents the percentage of the vertex number of the simpli£ed model to that of the original model. No bit error occurred until the simpli£cation reached 40-50%, and over 90% of the watermark remained until the model was simpli£ed to 21%. When the vertex randomization was performed, all the vertices $\mathbf{v}$ were added to $\alpha \times \text{uniform}() \times \mathbf{v}$. The modulation factor α was 0.02, which was also varied to create a bit error, while uniform() was a uniformly random function of $[-0.5 \; 0.5]$. In table I, 90% of the watermark remained. For the cropping attacks, all vertices with an x coordinate value over $\max_x/8$ were cropped, where $\max_x$ was the maximum x coordinate value. All the watermarks could be extracted with no error. The rotation attack also had no affect on the proposed model. The models attacked by simpli£cation, random noise, rotation, and cropping are

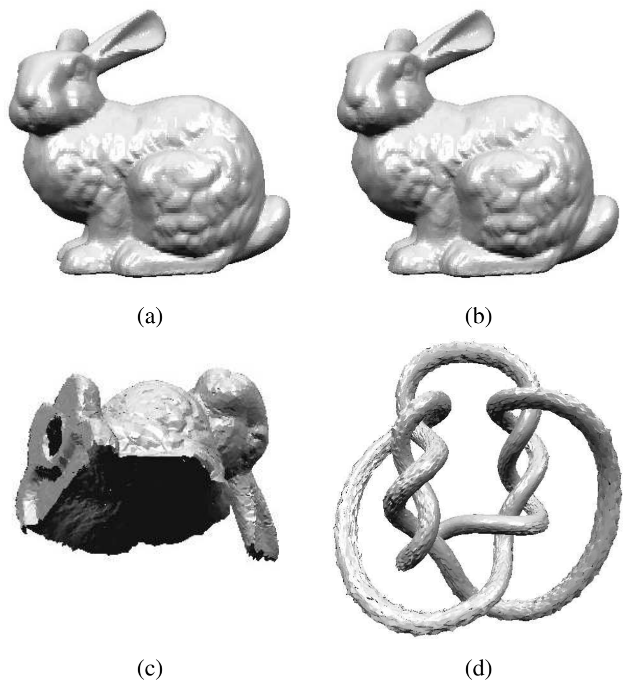

Fig. 3. (a) Original Stanford bunny, (b) watermark-embedded Stanford bunny, (c) simpli£ed, cropping, and rotation, and (d) knots added vertex randomization.

Table I. Performance results of the robustness against various attacks.

Test model	Attacks		BER
Stanford bunny (35,947 vertices, 69,451 connectivities)	Mesh simplify	35.28% (12,684 vertices)	2%
		21% (7,582 vertices)	10%
	Vertex randomization (a=0.02)		4%
	Cropping (14,284 vertices)		2%
	35.28% simplify + Cropping + rotation (4,957 vertices)		14%
Knots (23,232 vertices, 46,464 connectivities)	Mesh simplify	56.5% (13,136 vertices)	4%
		40% (9,316 vertices)	8%
	Vertex randomization (a=0.02)		10%
	Cropping (10,455 vertices)		0%
	56.5% simplify + Cropping + rotation (6,050 vertices)		10%

shown in Fig. 3(c)-(d).

5. CONCLUSIONS

3D mesh watermarking was proposed based on POCS. As such, a binary watermark is embedded by modifying the sample means of r components, which is performed by iteratively projecting the model onto a robust constraint convex set and invisible constraint set. Experiments veri£ed that the watermarked mesh was both robust to various attacks and invisible.

6. ACKNOWLEDGEMENTS

This work was supported by Institute of Information Technology Assessment of Ministry of Information and Communication in Republic of Korea

7. REFERENCES

[1] Y. Yang, and N. P. Galatsanos, ”Projection-Based Spatially Adaptive Reconstruction of Block-Transform Compressed Images,” *IEEE Trans. on Image Processing*, vol. 4, no. 7, July 1995.

[2] Z. Sun, A. M. Tekalp, N. Navab, and V. Ramesh, ”Interactive Optimization of 3D Shape and 2D Corresponding Using Multiple Geometric Constraints via POCS,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, April 2002.

[3] R. Ohbuchi, S. Takahashi, T. Miyazawa, and A. Mukaiyama, ”Watermarking 3D Polygonal Meshes in the Mesh Spectral Domain,” in *Proc. Graphics Interface 2001*, pp. 9-17, 2001.

[4] E. Praun, H. Hoppe, and A. Finkelstein, ”Robust Mesh Watermarking,” *Proc. SIGGRAPH ’99*, pp. 49-56, 1999.

[5] O. Benedens, ”Geometry-Based Watermarking of 3D Models,” *IEEE CG&A*, pp. 46-55, Jan./Feb., 1999.

[6] K. Kwon, S. Kwon, S. Lee, T. Kim, and K. Lee, ”Watermarking for 3D Polygonal Meshes Using Normal Vector Distributions of Each Patch,” *IEEE International Conference on Image Processing*, vol. 2, pp.499-502, Sep, 2003.

[7] T. Kanai, MeshToSS Version 1.0.1, http://graphics.sfc. keio.ac. jp/MeshToSS/indexE.html.