Journal of Zhejiang University SCIENCE
ISSN 1009-3095
http://www.zju.edu.cn/jzus
E-mail: jzus@zju.edu.cn

JZUS

Watermarking on 3D mesh based on spherical wavelet transform*

JIN Jian-qiu (金剑秋)[†1,2], DAI Min-ya (戴敏雅)[1], BAO Hu-jun (鲍虎军)[1], PENG Qun-sheng (彭群生)[1,2]

([1]*State Key Laboratory of CAD&CG, AI Institute, Zhejiang University, Hangzhou 310027, China*)

([2]*Department of Mathematics, Zhejiang University, Hangzhou 310027, China*)

[†]E-mail: jqjin@cad.zju.edu.cn

Received Apr. 16, 2003; revision accepted Nov. 12, 2003

Abstract: In this paper we propose a robust watermarking algorithm for 3D mesh. The algorithm is based on spherical wavelet transform. Our basic idea is to decompose the original mesh into a series of details at different scales by using spherical wavelet transform; the watermark is then embedded into the different levels of details. The embedding process includes: global sphere parameterization, spherical uniform sampling, spherical wavelet forward transform, embedding watermark, spherical wavelet inverse transform, and at last resampling the mesh watermarked to recover the topological connectivity of the original model. Experiments showed that our algorithm can improve the capacity of the watermark and the robustness of watermarking against attacks.

Key words: Mesh watermarking, Spherical wavelet transform, Spherical parameterization

Document code: A **CLC number:** TP309

INTRODUCTION

The rapid growth of digital media over Internet provides everybody with the facility of easy access, copy, edit and distribution of digital contents such as electronic documents, images, sounds and videos. There is urgent demand for techniques to protect the copyright of the original digital data and to prevent unauthorized duplication or tampering. Digital watermarking or data hiding is one solution for the copyright protection of digital data.

Digital watermarking is a process by which a user-specified signal (watermark) is hidden or embedded into another signal (cover data). Efforts on digital watermarking have been traditionally concentrated on media data such as audio, still image, and video. Recently, the increased popularity of three-dimensional mesh data has prompted research applying this technique to 3D models. In this paper we will describe a new algorithm of watermarking on meshes.

Related works on the watermarking 3D models

The pioneer works of watermarking 3D models are due to Dr. Ohbuchi *et al*.(1997; 1998). In his papers, the watermark is embedded into polygonal mesh data by modifying either the vertex coordinates, the vertex topology, or both. In order to resist attacks of geometrical transform, Ohbuchi *et al*. made use of embedding primitives, which are invariant under certain class of geometrical transformation. The embedding primitives can be geometrical or topological ones. The embedding primitives can be arranged when embedding and extracting watermark so that the watermarking algorithm can resist reordering vertex attack. Remeshing attack is not taken into consideration in this paper. In this case, original model data is not

*Project supported by the National Natural Science Foundation of China (Nos. 60021201, 60133020) and the National Basic Research Program (973) of China (No. 2002CB312101)

necessary during extracting watermark. It is a blind watermarking algorithm.

Emil *et al.*(1999) embedded the watermark during the construction of progressive meshes (PM). A set of scalar basis functions can be computed from the PM to determine the weight of the coordinate modification of vertices. Their methods included model registration during watermark detection, therefore their algorithm can resist attacks of vertices reordering, remeshing, geometrical transformation. But its accuracy of watermark detection depends on the accuracy of model registering.

While the algorithms mentioned above embed the watermark in the spatial domain, there are other classes of some 3D model watermarking algorithms which operate in the frequency domain. Kanai *et al.* (1998) decomposed a 3D polygon mesh into a multiresolution representation by performing lazy wavelet transform proposed by Lounsbery *et al.*(1993). The vertices of the coarsest shape of the model are unchanged under the linear wavelet transform, and only wavelet coefficients are modified for watermark embedding. As the embedding is not global, its watermark capacity is strictly limited, and the algorithm cannot effectively resist the attack of mesh simplification. In Eurographics 2002, Ohbuchi *et al.*(2002) also proposed a frequency domain approach to watermark 3D shapes. In the process, the mesh is segmented first into some patches, and then for each patch a spectral analysis is conducted; the watermark information is finally embedded into the frequency domain at the modulation step. Nevertheless, the patch generation step cannot be performed automatically, and manual interaction is needed. The remeshing attack countered by this method necessitates a mesh alignment step during the watermark extracting. The watermark extracting process contains mesh alignment which was also used by Emil *et al.*(1999).

Note that none of the above-mentioned watermarking algorithms exploits the property of HVS (Human Vision System). Motivated by many digital image watermarking algorithms, we present a new mesh watermarking algorithm based on spherical wavelet transform (Schröder and Sweldens, 1995). We decompose the mesh to get a multiresolution representation consisting of approximation part and several detailed parts. Then by exploiting the different visual sensitivity on different frequency band, we embed the watermark globally and adaptively, which enhances the capacity and robustness of watermarking.

OVERVIEW OF OUR APPROACH

In order to make full use of the HVS property of different visual sensitivity on different frequency band, we must transform the mesh into a frequency domain. Fig.1 illustrates the outline of the proposed watermarking method. Our basic idea is to decompose the original mesh into some detailed parts and

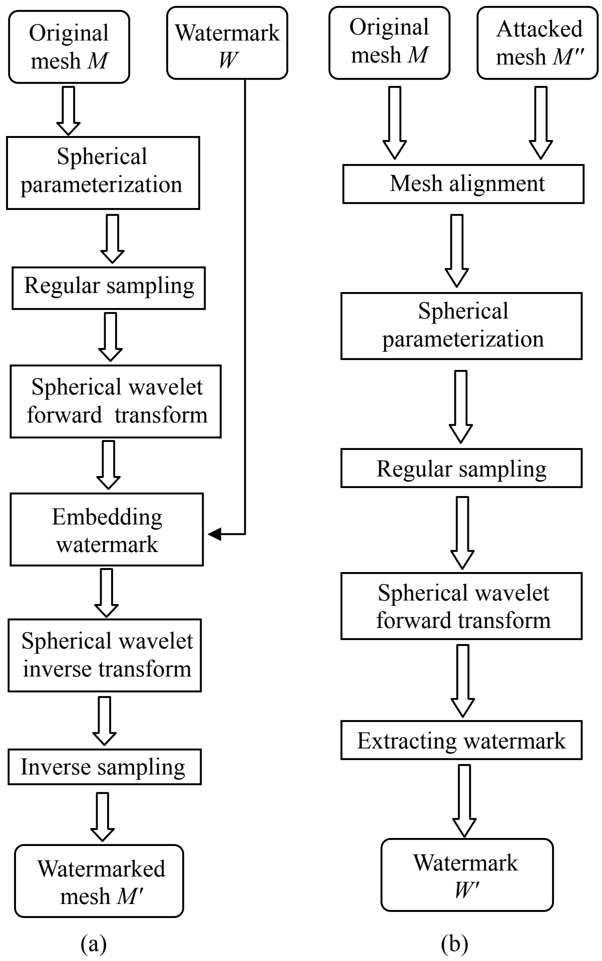

Fig.1 Overview of our approach
(a) Embedding; (b) Extracting

an approximation part by using sphere wavelet transformation proposed by Schröder and Sweldens (1995). Then watermark was embedded into both the detailed parts and the approximation part. Sphere wavelet transformation requires that the mesh must have the following properties: (1) Each vertex of the mesh must have its sphere coordinate. (2) The mesh be uniformly sampled. The first property means that we must first perform global sphere parameterization on the original mesh. So our method consists of the following steps:

1. Watermark embedding process:

Step 1: Perform global spherical parameterization. Spherical parameterization is mapping a mesh into a sphere such that the 3D model can be defined as spherical signals. This step requires that the mesh is homeomorphic to sphere. We will discuss this step in Section 3.

Step 2: Perform uniform spherical sampling. Spherical wavelet transform requires geometrical signals to be uniformly sampled over the sphere. We discuss this step with Step 1 in Section 3.

Step 3: Perform sphere wavelet forward transform. This step and Step 5 are inverse operations to each other; and they are discussed in detail in Section 4.

Step 4: Embed watermark.

Step 5: Perform inverse spherical wavelet transform.

Step 6: Resample the watermarked mesh to recover the topology connectivity of the original model.

2. Watermark detecting process:

The countering of the remeshing attack on the watermarked model requires a step of alignment or model registration. The steps of watermark detection are as follows.

Step 1: Perform mesh alignment.

Step 2: Perform Step 1 and Step 2 of watermark embedding process on both the original mesh and the watermarked mesh.

Step 3: Perform spherical wavelet forward transform on the two outputs of the previous step.

Step 4: Extract watermark information and compare the results.

GLOBAL SPHERICAL PARAMETERIZATION AND SAMPLING

Parameterization is crucial to many applications such as texture mapping, morphing and geometric signal processing. Several methods were developed for parameterization over the unit sphere (Alexa, 2002; Zhou *et al.*, 2002; 2003; Kent *et al.*, 1992; Haker *et al.*, 2000; Quicken *et al.*, 2000). We use the algorithm developed by Zhou *et al.*(2002). Here, we describe it briefly as follows. Readers looking for more details can refer to Zhou *et al.*(2002; 2003).

As illustrated in Fig.2, the method involves two steps:

1. Generate a progressive mesh representation

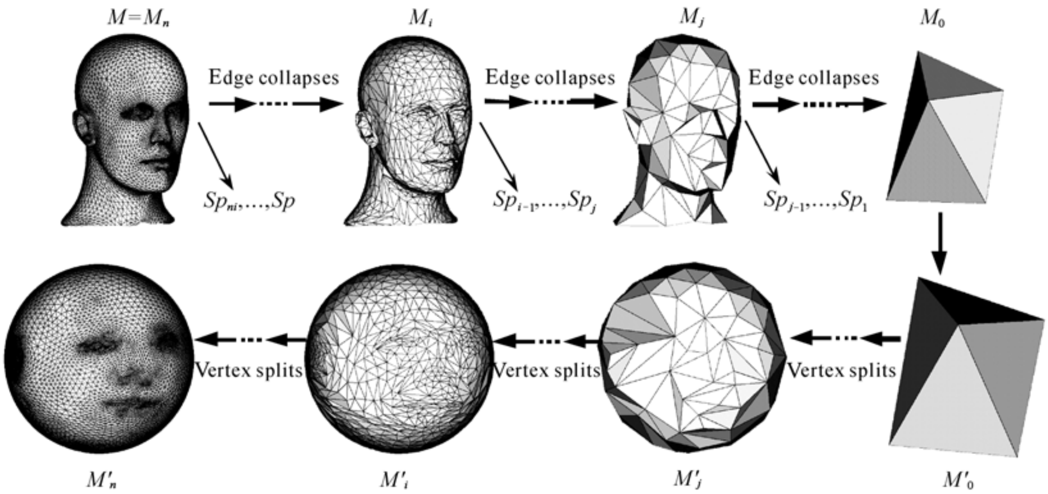

Fig.2 Global spherical parameterization

with local parameterization information. Edge collapse operation is iteratively performed until the mesh is simplified into a convex polyhedron. For each edge collapse, the two decimated vertices are parameterized over the resultant simplified mesh. The local parameterization information is recorded in the PM.

2. Start with the initial spherical mesh yielded by projecting the base mesh in the PM onto the unit sphere; the sequence of vertex split operations in the PM is performed progressively. For each vertex split, the two split vertices are positioned on the unit sphere using the recorded connectivity and local parameterization information. The procedure of edge collapse with local parameterization is in Fig.3.

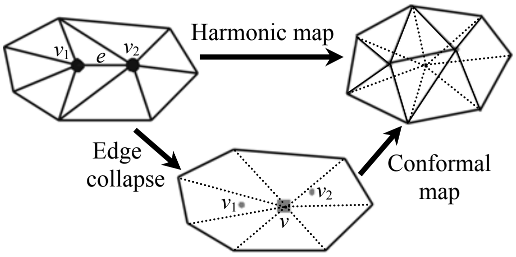

Fig.3 Edge collapse with local parameterization

Sampling and precision control

We obtained the geometrical signal on the unit sphere in the previous section. In order to perform spherical wavelet transform over the geometrical signal we need to sample the signals regularly over the sphere. As illustrated in Fig.4, we first perform recursive 1-split-to-4 subdivision of the tetrahedral base shape (or other Platonic Solid) as used by Schröder and Sweldens (1995), then we sample the signals at the vertices of the subdivision spherical mesh. In practice, we wish that the generated regular mesh approximates the original mesh with a given tolerance ε. Let M be the original mesh and SM is the sampled mesh. We perform inverse sampling an SM to get mesh M'. The 1-to-4 subdivision is recursively performed until the following equation is satisfied

$$d(M, M') = \max_{i \in v_M} \| v_i' - v_i \| < \varepsilon ,$$

where ε is a user-specified error threshold, v_i and v_i' are vertices on M and M' respectively. In our current implementation, the initial subdivision level is five.

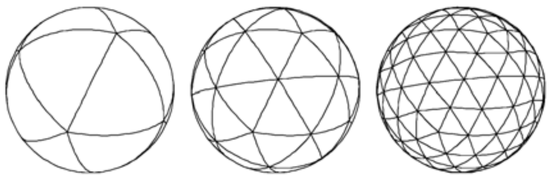

Fig.4 Spherical meshes subdivision. The subdivided meshes are used for sampling

SPHERICAL WAVELET TRANSFORM

Wavelets are basis functions which represent a given function at multiple levels of detail. Due to their local support in both spatial domain and frequency domain, they are suited for sparse approximations of functions. In the real number field, wavelets are defined as the dyadic translation and dilation of one particular, fixed function and in multi-dimension Euclidean space tensor-production wavelet is usually adopted. But in $\mathcal{S}^2$ space, the translation invariance is no longer true, and wavelet cannot be represented as translation and dilation of one function. In this paper, we adopt the spherical wavelet proposed by Schröder and Sweldens (1995) and select the linear lifting wavelet transform in particular. We describe it briefly as follows: For general wavelet transform,

Analysis:

$$\lambda_{j,k} = \sum_{l \in K(j)} \tilde{h}_{j,k,l} \lambda_{j+1,l} , \gamma_{j,m} = \sum_{l \in M(j)} \tilde{g}_{j,m,l} \lambda_{j+1,l} \qquad (1)$$

Synthesis:

$$\lambda_{j+1,l} = \sum_{k \in K(j)} h_{j,k,l} \lambda_{j,k} + \sum_{m \in M(j)} g_{j,m,l} \gamma_{j,m} \qquad (2)$$

When $\lambda_{n,\bullet}$ (n is finest resolution level) is given, we can recursively perform the above analysis process (forward transform) to get $\gamma_{j,\bullet}$ the wavelet coefficients at the current level, and the coarsest approximation part $\lambda_{n-i,\bullet}$ after performing the decomposition i times. Similarly, if we have $\lambda_{n-i,\bullet}$ and $\gamma_{j,\bullet}$ (j=$n-i$, $n-i$+1, ..., n−1), we can perform the synthesis process (inverse transform) recursively to get the $\lambda_{n,\bullet}$. Different h, $\tilde{h}$, g, $\tilde{g}$ denote different wavelet basis function. In Euclidean space we have $h_{j,k,l}= h_{l-2k}$ (the same with $g, \tilde{g}$) , but in general manifold they are dependent on scale and position. Now we concentrate on spherical wavelet transform. In Eqs.(1) and (2), $M(j)$ and $K(j)$ are abstract sets; we will make these index sets concrete in sphere. A diagram is given in Fig.5. The mesh including dashed edges in the figure is assumed as resolution j+1 level. $K(j)$ denotes the point set of the intersection points of the solid lines and $M(j)$ denotes the set of the intersection points of the dash lines. And their union $K(j)\cup M(j)=K(j+1)$. We will compute the λ_j and γ_j approximation part and detailed part, by single decomposition in the neighborhood of m. Three transform methods: Lazy, Linear, Linear lifting are briefly investigated.

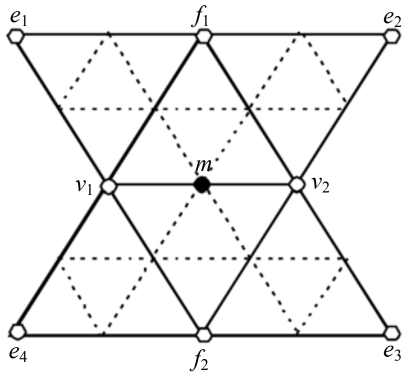

Fig.5 Spherical wavelet transform: neighbors used in our bases

Lazy: The Lazy wavelet does nothing but subsampling. The result of the analysis and synthesis steps become:

Analysis: $\lambda_{j,k}=\lambda_{j+1,k}$, $\gamma_{j,m}=\lambda_{j+1,m}$,

Synthesis: $\lambda_{j+1,m}=\gamma_{j,m}$, $\lambda_{j+1,k}=\lambda_{j,k}$,

where $k\in K(j)$, $m\in M(j)$.

Linear: The scaling coefficients (approximation part) are subsampled and kept unchanged, the coefficients of finer resolution are predicted by linear interpolation:

Analysis: $\lambda_{j,k}=\lambda_{j+1,k}$

$$\gamma_{j,m} = \lambda_{j+1,m} - 1/2(\lambda_{j+1,v_1} + \lambda_{j+1,v_2})$$

Synthesis: $\lambda_{j+1,k}=\lambda_{j,k}$

$$\lambda_{j+1,m} = \gamma_{j,m} + 1/2(\lambda_{j,v_1} + \lambda_{j,v_2})$$

Linear lifting: Update the scaling coefficients by using the wavelet coefficients of linear wavelet transform to assure that the wavelet has at least one vanishing moment:

Analysis: $\gamma_{j,m} = \lambda_{j+1,m} - 1/2(\lambda_{j+1,v_1} + \lambda_{j+1,v_2})$

$$\lambda_{j,k} = \lambda_{j+1,k} + 1/s_j(\textstyle\sum_{m\in\mathcal{N}(k)}\gamma_{j,m})$$

Synthesis: $\lambda_{j+1,k} = \lambda_{j,k} - 1/s_j(\textstyle\sum_{m\in\mathcal{N}(k)}\gamma_{j,m})$

$$\lambda_{j+1,m} = \gamma_{j,m} + 1/2(\lambda_{j,v_1} + \lambda_{j,v_2})$$

where s_j must be chosen to ensure that the resultant wavelet has a vanishing moment (Schröder and Sweldens, 1995). In our experiments, tetrahedron was chosen as the original subdivision mesh. In this case s_1=1/5, s_2=1/6, and $s_j\approx$1/8 when $j\geq$5. In our watermarking algorithm, linear and linear lifting are adopted. And the two transform methods were compared in our experiment which will be shown in Section 7.

Other transform methods: in Schröder and Sweldens (1995), Quadratic, Butterfly transform methods are also given. In these methods, e_i, f_i shown in Fig.5 must be used.

EMBEDDING WATERMARK

In the previous section, we decomposed the geometric signal of the approximation part and detailed parts. Large coefficients of detailed parts correspond to some local bumpiness. Since the human vision is not sensitive to minor geometric modification in bumpy areas, we can embed sufficient energy of watermark into these areas. On the other hand, the coefficients of low frequency determine the rough shape of the 3D mesh and they possess larger magnitudes than those of the high frequency. We can adaptively embed more watermark information into the approximation part than in the detailed parts. The above adaptive watermark embedding can be described by the following mathematical formula.

$$v_i'^j = v_i^j + g_j f(v_i^j) w_i$$

where v_i^j is the ith vertex of M and belongs to band j. w is watermark. Function $f(\bullet)$ computes the weight of embedding intensity, which is related with the band j. And g_j is used to control the global intensity of the watermark and is only related with band j. In our current implementation, function f is

$$f(v_i^j) = \begin{cases} (1/j)\sqrt{v_i^j} & \text{if band } j \text{ is one of detailed parts} \\ \log v_i^j & \text{if band } j \text{ is an approximation part} \end{cases}$$

MESH ALIGNMENT AND WATERMARK DETECTION

In modeling or rendering, it is common to apply a geometric transform such as: translation, rotation, scaling, reordering vertex, remeshing and simplification on a 3D model. Such a transform often breaks information synchronization. In order to extract the watermark, we must perform a mesh alignment (mesh registration) in the watermark extraction process.

Several methods were developed for mesh alignment (Besl and Mckay, 1992; Chen and Medioni, 1992). In this paper, we adopt the algorithm developed by Chen and Medioni (1992). It is an iterative algorithm. In the algorithm, only rotation and translation are considered, but it is easy to incorporate one additional degree of freedom: unform scaling of the mesh into the alignment process. Human interaction is required to provide a reasonable initial condition in most cases.

Note that after the mesh alignment, remeshing is not required because we will resample the mesh after the spherical parameterization.

After mesh alignment, we conduct a uniform spherical sampling and then spherical wavelet forward transform on the two meshes. We then extract the watermark through comparison of these two meshes. There are two ways to determine whether the extracted watermark corresponds to the original watermark: the first, to calculate the correlation coefficient between them, and the second, to count the total bit error rates. We use the latter in our experiments. Table 1 reports our experiments results.

EXPERIMENTS AND RESULTS

In our implementation, we took {0,1} sequence which was generated randomly as watermark conducted wavelet decompositions of linear and linear lifting wavelets respectively. We tested the robustness of our algorithm under several attacks: reordering vertex, mesh simplification and noise imposition attack, as illustrated in Fig.6. The initial results showed that linear lifting is better than linear in perceptual invisibility and capacity of embedding. This is because when performing linear lifting, the error resulting from watermark embedding can be better spread around neighboring vertexes compared with the linear transform. The evidence become apparent when watermark is embedded into scaling coefficients. Table 1 shows the ability of resiliency against some attacks: reordering vertex, mesh simplification, noise imposi-

Table 1 Our experiments results: Bit error rates were used to measure the robustness of the algorithm

Attack	Bunny		Venus		Mannu.	
	Linear	Linear lifting	Linear	Linear lifting	Linear	Linear lifting
Reordering	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
Simplification 1/4	0.135^* 0.087	0.065	0.184^* 0.104	0.081	0.167^* 0.085	0.072
Simplification 1/8	0.326^* 0.165	0.107	0.383^* 0.173	0.134	0.302^* 0.221	0.108
Noise 0.3%	0.063	0.042	0.076	0.045	0.067	0.038
Noise 0.6%	0.125	0.103	0.128	0.092	0.143	0.088

* denotes the watermark was only embedded into the detailed parts

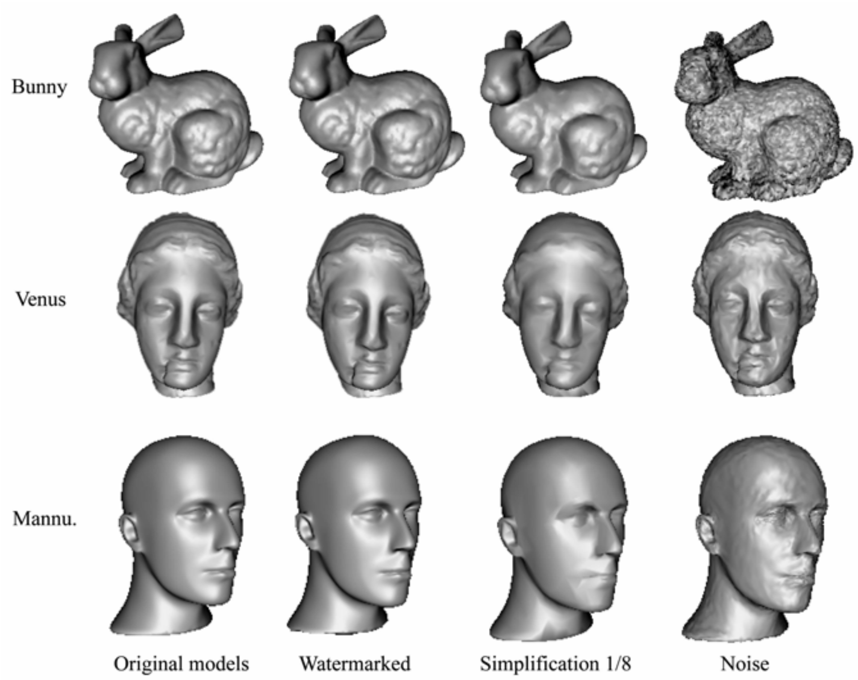

Fig.6 Our experiments results. 0.6% noise for Bunny, 0.3% noise for others

position. In the table we also show the comparison result of linear and linear lifting. Table 1 clearly shows that the linear lifting method performs a little more robust than the linear method during resistance against those attacks. Those attacks are discussed roughly below.

Reordering vertex

Reordering vertex is an attack when only the order of vertexes is modified and no geometry coefficients and topological connectivity are modified. It is a kind of lossless attack. During the watermark extraction, before spherical wavelet decomposition, we must align the original mesh and the attacked mesh. If we get the precise alignment, the watermark can be perfectly extracted. In practice, the alignment process cannot perform perfectly and small error is allowed. Our experiments showed the proposed approach can extracted successfully the watermark from the meshes suffering vertex reordering attack.

Mesh simplification

The mesh simplification algorithm adopted by us was proposed by Zhou *et al*.(2000). The result of our experiments showed that the watermark cannot be detected if no watermark is embedded into scaling coefficients. And as we expected, the result of linear lifting is better than that of linear. Furthermore, the difference in robustness is prominent when simplification rate is high.

Noise imposition

Noise imposition is a kind of attack by adding random noise to vertex coordinates. The bit error rates (BER) of the watermark are also evaluated as above. The results in Table 1 show good resiliency against noise imposition.

References

Alexa, M., 2002. Recent advances in mesh morphing. *Computer Graphics Forum*, **21**(2):173-196.

Besl, P.J., Mckay, N.D., 1992. A method for registration of 3D shape. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **14**(2):239-256.

Chen, Y., Medioni, G., 1992. Object modeling by registrat ion of multiple range images. *Image and Vision Comptutting*, **10**(4):145-155.

Emil, P., Hugues, H., Adam, F., 1999. Robust Mesh Watermarking. Proceedings of SIGGRAPH 1999, Computer Graphics Proceedings, Annual Conference Series, ACM, p.49-56.

Haker, S., Angenent, S., Tannenbaum, A., Kikinis, R., Sapiro, G., Halle, M., 2000. Conformal surface parametrization for texture mapping. *IEEE Transactions on Visualization and Computer Graphics*, **6**(2):181-189.

Kanai, S., Date, H., Kishinami, T., 1998. Digital Watermarking for 3D Polygons Using Multiresolution Wavelet Decomposition. IFIP/WG5.2/Geo-6'98.

Kent, J.R., Carlson, W.E., Parent, R.E., 1992. Shape Transformation for Polyhedral Objects. Proceedings of SIGGRAPH 1992, Computer Graphics Proceedings, Annual Conference Series, ACM, p.47-54.

Lounsbery, M., DeRose, T., Warren, J., 1993. Multiresolution Surface of Arbitrary Topological Type. Tech. Rep., Department of Computer Science and Engineering, University of Washington.

Ohbuchi, R., Masuda, H., Aono, M., 1997. Watermarking 3D Polygonal Models. Proceedings of ACM Multimedia'97, ACM, p.261-272.

Ohbuchi, R., Masuda, H., Aono, M., 1998. Watermarking three dimensional polygonal models through geometric and topological modifications. *IEEE Journal on Selected Areas in Communication*, **16**(3):551-560.

Ohbuchi, R., Akio, M., Shigeo, T., 2002. A frequency-domain approach to watermarking 3D shapes. *Computer Graphics Forum*, **21**(3):373-382.

Quicken, M., Brechbühler, C., Hug, J., Blattmann, H., Székely, G., 2000. Parameterization of Closed Surfaces for Parametric Surface Description. CVPR, p.354-360.

Schröder, P., Sweldens, W., 1995. Efficiently Representing Function on the Sphere. Proceedings of SIGGRAPH 1995, Computer Graphics Proceedings, Annual Conference Series, ACM, p.161-172.

Zhou, K., Zhang, M., Pan, Z., Shi, J., 2000. A new mesh simplification algorithm for colored or textured polygonal models. *The International Journal of Virtual Reality*, **16**(1):57-63.

Zhou, K., Bao, H., Shi, J., 2002. A unified framework for digital geometry processing. *Chinese Journal of Computers*, **25**(9):904-909 (in Chinese).

Zhou, K., Bao, H., Shi, J., 2003. 3D surface filtering using spherical harmonics. *Computer Aided Design*, to appear.