Statistical Robust Watermarking for 3D Mesh Models Based on Salient Points

Nassima Medimegh*, Samir Belaid†, Mohamed Atri* and Naoufel Werghi‡
*Electronics and Microelectronics Laboratory, Faculty of Sciences, University of Monastir, TUNISIA
Email: Medimegh_nassima@yahoo.fr
Email: mohamed.atri@fsm.rnu.tn
†MARS Research Unit UR11ES57, Department of Computing, University of Monastir, TUNISIA
Email: Samir.Belaid@fsm.rnu.tn
‡ Khailfa University, Abu Dhabi, UAE
Email: Naoufel.Werghi@kustar.ac.ae

***Abstract*—We propose in this paper, a robust blind watermarking algorithm for 3D mesh models by embedding bits of signature in distributions of vertex norms. Using the Fast Marching Method (FMM), we define regions on the mesh according to a reference location. This robust source location are selected by the salient point detector based on Auto Diffusion Function (ADF). We show that robustness against cropping and other common attacks is achieved. Due to stability of salient points, we can retrieve the watermarked region and extract the watermark. Also, the proposed method has provided a minimal surface distortion in the embedding process.**

***Index Terms*—3D watermarking; salient point detector; Fast marching method; statistical method.**

I. Introduction

With the proliferation of digital multimedia across the web, protecting the copyright and the integrity of the numerical data is become a necessity. Watermarking which is the process of embedding signature into digital media, for example : videos, images and 3D objects, is considered as an efficient solution for copyright protection.

With the advancement of technologies of 3D acquisition devices and the computer-aided Design (CAD) technology, 3D objects become among the most important parts of multimedia content. This models are used in many sectors spanning, Manufacturing, Medicine and Entertainment, etc. This resulted the requirement of new paradigms and approaches that can consider the difficult aspects of 3D models watermarking, and especially 3D triangular mesh.

Since its introduction Ohbuchi et al. [1], there have been various tries to ameliorate the performance of watermarking such as robustness and invisibility. Many 3D model watermarking methods have been developed. This variety is classified into two categories: Fragile watermarking methods allow to check if a model is changed and robust watermarking methods are destined to support the mesh attacks. The watermarking can be non-blind or blind, depending on the necessity of the original model in the watermark detection process. In this work, we present an approach for 3D mesh watermarking robust and blind.

Watermarking algorithm can be in spatial or spectral domains. Transforms methods are also divided into two classes: spectral methods and multiresolution methods.

Generally, spectral methods insert the signature into certain coefficients of harmonic or multiscale transform. The method of Cayre et al. [2] employed piece-wise Laplacian decomposition to insert watermark on the low and medium frequency. Wang et al. [3] combined manifold harmonic basis and elliptic curve digital watermark algorithm in a robust non-blind watermarking scheme. Chen et al. [4] presented a watermarking approach using the biorthogonal non Uniform B-Spline Wavelets. This methos is non-blind.

The spatial methods can either act on the topology or on the geometry of the model. The method of Jing et al. [5] is non-blind. They used the mean of normal in order to make new coordinates system. Recently, Rolland-Neviere X et al. [6] modified the vertex positions along the radial directions. Mao et al. [7] subdivided the triangles and they embedded the signature in the newly added vertices. This approach is topological and it is robust to affine transformation.

Cayre and Macq [8] proposed high capacity steganographic methods. They embed signature bit according to projecting a triangle vertex on the opposite edge. In the same concept, Werghi et al. [9] selected a sequence of facets using the Ordered Rings Facets structure [10]. Then, they applied the watermarking method of Cayre and Macq [8] on this facets.

Watermarking methods can also be statistical methods. This category extract the watermark by a statistical test. Cho et al. [11] presented blind and robust statistical methods. They embed a watermark by modifying the mean or the variance of normalized distributions of the vertices norms depending on the watermark bit. Despite its robustness, these algorithms cause visible artifacts on the 3D models surface. Hu et al. [12] proposed a similar histogram-based method by using quadratic programming. This method is more robust to Gaussian noise, compared with Cho's method. After modifying the statistics of the distances, Luo and Bors [13] search the best position of some vertex by applying the Quadratic Selective vertex Placement algorithm. Zhan et al. [14] embedded the watermark into vertex bins by modulating the mean fluctuation values of the bins.

A general review of 3D mesh watermarking is established in [15]. Experiments proved that histogram-based methods can be robust to usual watermarking attacks excluding cropping. In this paper, we presented a histogram-based watermarking method for 3D models aiming to ameliorate robustness to a diversity of attacks. We applied similar approach as in [11] in order to change the distribution of vertex norms. The proposed aproach has the following stages: extracting the references locations and the 3D mesh surface divided into geodesic Voronoi regions. From each region,the vertices are used for embedding a watermark bits. For that, we modified the distribution mean of vertex norms. Compared to Cho's watermarking method [11], The proposed watermarking scheme is more resistant to attacks. Experiments show that the employ of the robust feature points and the segmentation of mesh make the proposed method resilient against cropping attacks.

The rest of the paper is arranged as follows: Section II presents the preliminaries and describes the watermark embedding and extraction method in detail. Some experimental results for our method are also exposed and discussed in Section III.Concluding remarks are briefly discussed in Section IV.

II. The Watermarking Algorithm

The steps of proposed watermark embedding method are: firstly, we define the references locations, secondly, we segment the object surface into geodesic Voronoi cells and finally, we embed statistically the watermark bits into each region.

A. Preliminaries

1) Laplace-Beltrami operator: Δ designate the Laplace-Beltrami differential operator defined on manifold surface. H is the eigenfunction associated to the eigenvalue λ. Consider the formula for the discrete Laplace-Beltrami written using the Finite Element Method [16] :

$$-Qh = \lambda Dh \tag{1}$$

h represents the vector space of dimension M, where M vertices in the mech. The lumped matrix D is given by :

$$D_{ii} = \frac{1}{3} \sum_{t \in N_t(i)} |t| \tag{2}$$

Where $N_t(i)$ is the ensemble of neighbouring facets from vertex v_i and the stiffness matrix Q is given as follows:

$$Q_{i,j} = \begin{cases} Q_{i,j} = \frac{\cot(\beta_{i,j}) + \cot(\beta'_{i,j})}{2} \\ Q_{i,i} = -\sum_j Q_{i,j} \end{cases} \tag{3}$$

The angles $\beta_{i,j}$ and $\beta'_{i,j}$ appearing in this formula are the opposite angles to the edge $v_i v_j$.

2) Heat Kernel Diffusion: In an unstable condition, the heat diffusion process represents the progress of a function onto a surface. It is patterned by the method Heat Kernel HK. $K_t(x, y)$ is presented as the probability that the point y is achieved from the point x at time t. Here x and y are two points on surface. The HK is based on the basis of eigenfunctions correspond to different eigenvalues of Laplace-Beltrami operator. $K_t(x, y) \in$ (M×M×R$^+$). It is defined as follows:

$$K_t(x, y) = \sum_{n=0}^{\infty} \exp(-\lambda_n t) h_n(x) h_n(y) \tag{4}$$

B. Our Watermarking Method

The proposed method consists on embedding watermark bits statistically and locally. Considering the feature point location as reference, the 3D model is divided into region. So we define the local region by using the fast marching algorithm FMM which is described in the following sections.

1) Robust Feature point extraction: So as to define a robust source location, we just used the salient point detector proposed by Haj-Mohamed and Belaid [17].

Salient points on 3D shapes are not altered by scaling, rotation, additive noise, articulation, and deformation. In addition, those points must have a distinct locality, and must be stable at all cases of a model. Haj-Mohamed and Belaid [17] developed an unsupervised and automatic 3D salient point detector. The proposed detector is founded on Auto Diffusion Function ADF introduced by Gbal et al. [18].

This scalar function is represented as a linear sequence of the Laplace-Beltrami Operator eigenfunctions. Here, we use the parameter t to supervise the number and the capacity of the extracted features, just we varied the parameter t.

$$\begin{aligned} ADF_{\frac{t}{\lambda_2}} &= K(x, x) \\ &= \textstyle\sum_i \exp(-t \frac{\lambda_t}{\lambda_2}) h_i^2(x) \end{aligned} \tag{5}$$

As a main advantage of the ADF, its local extrema prove to be natural interest points. Figure 1 illustrates examples of extracted feature points using the ADF function.

It has been also showed that the extracted features are invariant to isometry or non-rigid deformations, micro holes, occultation, scaling, additive Gaussian noise and Laplacian Smooth [17].

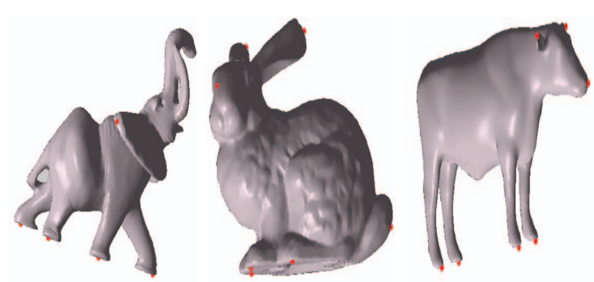

Fig. 1. Example of detected salient points.

The proposed method consists on embedding watermark locally. Referring to the source location extracted, the 3D object is split into region. So we define the local region by using the fast marching algorithm FMM which is described in the following sections.

2) Region extraction: Using the fast marching applied on a triangulated mesh, the geodesic distance is computed from a set of input feature points, and then we segment the surface into geodesic Voronoi cells.

a) Geodesic distance: The geodesic distance is determined by the length of the shortest path joining two locations on the object mesh surface M. The fast marching algorithm is used for this purpose [19], [20].

In fact, the geodesic distance T_o(x,y) between the endpoints x and y is the shortest length over all continuous paths determined by the geodesic curve γ(t).

$$T_o(x,y) = \min_{\gamma(t)\in o} \int_0^P \sqrt{\gamma'(t)^T H(\gamma(t))\gamma'(t)}dt \qquad (6)$$

Here, γ(0)=x, $\gamma(P)$=y, γ'(t) are, respectively the local derivative of the parametric curve whereas H(-) represents an intrinsic metric.

As the geodesic distance is an integrative function, it resists to smoothing, noise and resampling.

b) Geodesic Voronoi Diagram: Let P = p_1, p_2, , p_m be an ensemble of points on mesh M. The Voronoi cell VC(p_i) of the site pi includes all points whose distance to p_i is less than or equal to their distance to any other site [21], i.e., VC(p_i) = {q $\in$ M|d(p_i, q)$\leq$d(p_j , q), for all i $\neq$ j}. So, the geodesic Voronoi diagram (GVD) of P is the union of all Voronoi cells, GVD(P) = {VC(p_1), VC(p_2), , VC(p_m)}.

Fig. 2. Examples of Voronoi segmentation on 3D models.

3) Watermark Embedding: In the following, we consider the same notations and terminologies as in [11] for the raison that our research work follows theirs. In fact, this method embeds watermark at the cadence of one bit per bin. Here we change the mean value of each set depending on the assigned signature bit. So, by using a histogram mapping function, we modify norms of the vertex in each bin. Afterwards, we will describe the data embedding process in detail.

Let us assume that we want to insert N bits of a watermark w into the vertices v_i of 3D mesh. First, the Cartesian coordinates of a a given vertex v_i (x_i ,y_i ,z_i) in the mesh are converted to a spherical coordinates (ρ_i,θ_i,φ_i). Second, the norms of the vertex vertex norms are distributed over N distinct bins based on their their magnitude. By using one bin to insert just one watermark we get a total of N embedding bits per selected region in the mesh. We obtained the largest (ρ_{max}) and the smallest (ρ_{min}) vertex norms according to:

$$\rho_{n,min} = \rho_{min} + \frac{\rho_{max} - \rho_{min}}{N}.n \qquad (7)$$

$$\rho_{n,max} = \rho_{min} + \frac{\rho_{max} - \rho_{min}}{N}.(n+1) \qquad (8)$$

Thus the bin B number n is defined as :

$$B_n = \{\rho_{n,j} | \rho_{n,min} < \rho_{n,j} < \rho_{n,max}\} \qquad (9)$$

Here $\rho_{n,min}$ and $\rho_{n,max}$ are lower and upper boundaries of the nth bin, and $\rho_{n,j}$ is the jth vertex norm in the nth bin.

The next step consist on normalizing the vertex norms of the bin number n to the interval [0, 1] by using:

$$\tilde{\rho}_{n,j} = \frac{\rho_{n,j} - \rho_{n,min}}{\rho_{n,max} - \rho_{n,min}} \qquad (10)$$

Here $\tilde{\rho}_{n,j}$ is the vertex norm of the vertex number j in the nth bin after normalization.

The watermarking acts here by perturbing $\tilde{\rho}_{n,j}$, to make the mean value of each bin via transforming vertex norms shifted via the histogram mapping function. The insertion of one bit is performed by modifying the mean of the histogram as follows:

$$\tilde{\mu}_n = \begin{cases} \frac{1}{2} + \alpha & if \quad w_n = +1 \\ \frac{1}{2} - \alpha & if \quad w_n = 0 \end{cases} \qquad (11)$$

With the strength parameters α, we can control the robustness and the transparency of watermark.

The iterative algorithm that embeds a watermark bit into a bin is described [21] as follows:

For embedding w_n=+1 into the nth bin:

1) Initialize the parameter k_n to one ;

2) Change normalized vertex norm by $\tilde{\rho}'_{n,j}$=$(\tilde{\rho}_{n,j})^{k_n}$

3) Calculate mean of transformed vertex norms by $\tilde{\mu}'_n$=$\frac{1}{M_n}\sum_{j=0}^{M_n-1}\tilde{\rho}'_{n,j}$

4) if $\tilde{\mu}'_n$ < (1/2) +α decrease $k_n(k_n = k - \Delta k)$ and return to step 2);

5) Replace normalized vertex norms with transformed norms using $\tilde{\rho}_{n,j}$=$\tilde{\rho}'_{n,j}$

6) end.

For embedding w_n=0 into the nth bin:

4) if $\tilde{\mu}'_n$ > (1/2) +α increase $k_n(k_n = k + \Delta k)$ and go back to 2);

After adding the distortions to the corresponding vertices, the vertex norms are converted back to their originals ones by:

$$\rho'_{n,j} = \tilde{\rho}'_{n,j}(\rho_{n,max} - \rho_{n,min}) + \rho_{n,min} \qquad (12)$$

The final step of watermark embedding process is to convert the spherical coordinates to Cartesian coordinates. These steps of embedding will be repeated for all region selected.

4) Watermark Detection: The watermark extraction procedure is blind. It is the same steps as in the embedding procedure. First, we detect the source location. From this points, we generate the geodesic Voronoi cells as described in previous Sections. Then, the watermarked mesh object is converted to spherical coordinates. In each region, the vertex norms are classified into bins and mapped onto the normalized between 0 and 1, and then, the mean of each bin is calculated and compared with $1/2$. The bit from the nth bin is obtained by:

$$w'_n = \begin{cases} +1 & if \tilde{\mu}" > \frac{1}{2} \\ 0 & if \tilde{\mu}" < \frac{1}{2} \end{cases} \qquad (13)$$

III. Experimental Results

The proposed method has been implemented on Matlab. To evaluate the proposed approach, it is applied on a different mesh models used usually in 3D watermarking application. Only three examples are presented here: Bunny (12942 vertices, 25736 faces), elephant (12471 vertices, 24950 faces) and cow (9511 vertices, 19018 faces) of a public database. Figure 3 shows the original models used. In the simulation, 64 bits are dedicated to the watermark code.

A. Surface distortion evaluation

A first desirable aspect of a watermarking method is a good visual imperceptibility of the watermark. To measure the quality distortion, we used Metro [22]. This method consists to calculate the Hausdorff Distance HD between the original mesh model and marked one. To measure the HD the following formula is used:

$$HD = max\{h(M_1), h(M_2)\} \tag{14}$$

Where M_1= (V, V') and M_2= (V', V), (V and V' represent respectively the original mesh and watermarked mesh).
$h(M_1)$= max{min(d(a,V'))}, a in V,
$h(M_2)$= max{min(d(b,V))}, b in V'.

We evaluated the visual effect of the mesh alterations produced by watermarking. Figure 4 presents the test models after watermarking. No differences can be observed between the original and marked mesh. In addition the low result of HD calculated between the mesh before and after embedding the watermark proves the good invisibility. We notice that the object surface distortion produced by our watermarking methodology is lower than that introduced by Cho's.

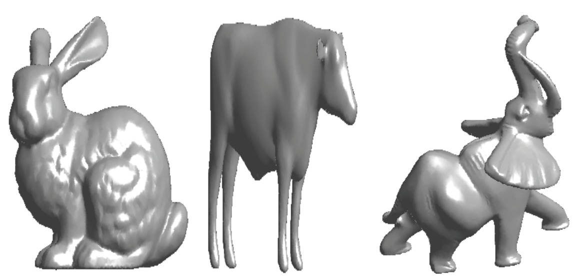

Fig. 3. 3D objects used in the experiments.

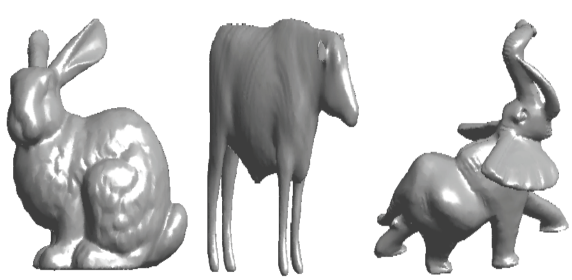

Fig. 4. Marked 3D objects.

TABLE I
Watermarked object distortion

	Bunny		Cow		Elephant	
	Our	Cho	Our	Cho	Our	Cho
HD ($\times 10^{-3}$)	0.73	0.74	5.8	5.8	3.5	7.8

B. Evaluation of Robustness

The most important criterion that must prove a watermarking method is robustness to attacks. In the following we test the robustness of the proposed method, we attack the watermarked object and try to extract the watermark from it. The robustness is measured by calculating the correlation between the original and the extracted signature.

$$corr = \frac{\sum_{n=0}^{N-1}(w_n - \bar{w})(w_n^{'} - \bar{w'})}{\sqrt{\sum_{n=0}^{N-1}(w_n - \bar{w})^2 \times \sum_{n=0}^{N-1}(w_n' - \bar{w'})^2}} \tag{15}$$

Where $\bar{w}$ is the average of the signature and the correlation corr in [-1, 1].

To evaluate the robustness, different attacks are tested. In the following we apply the attack: RST attacks (translation, scaling and rotation), Laplacian smoothing, additive noise, mesh quadratic metric simplification, and cropping to the watermarked model. We compared the proposed method with Chos watermarking algorithm [11]. The results of the evaluation are summarized in the table 2. From the experiment results, we can prove that our algorithm generally provide similar results of robustness as Cho et al. [11].

TABLE II
Experimental results against attacks: the maximum correlation obtained for the detected region

	Bunny		Cow		Elephant	
	Our	Cho	Our	Cho	Our	Cho
No attacks	0.94	1	1	1	0.94	1
RST	0.94	1	1	1	0.94	1
Smoothing	0.94	0.72	0.88	0.70	0.94	0.88
Noise 0.1%	0.94	0.85	0.94	0.97	0.94	0.78
Simplification 30%	0.88	0.94	0.94	0.50	0.76	0.90
Cropping	0.94	no	1	no	0.94	no

TABLE III
Number of detected regions

	Bunny	Cow	Elephant
No attacks	6	8	7
RST	6	8	7
Smoothing	4	5	6
Noise 0.1%	5	6	7
Simplification 30%	4	5	5
Cropping	3	7	5

The number of regions where we detect the watermark, are listed in table 3. We compute this number when the watermarked model is modified with different attacks. We note that, for every tested model, we extract the signature from many regions. Precisely in the case of cropping, although there were some patches are removed, the watermark could be extracted from the other patches.

As can be observed from the table 2, our method provide better results for the smoothing attack and slightly better for the other attacks. The proposed method is robust against partial geometric deformation, such as cropping. The reason is to embed the watermark in many regions which facilitate the detection of the signature after a cropping of the watermarked mesh.

IV. Conclusion

We have showcased a statistical 3D watermarking method using and embedding watermark into each region extracted from the mesh model using the FMM. We define the references locations and the 3D graphical object surface split into geodesic Voronoi region. A statistical method is employed for signature insertion, by changing the mean of distributions of vertex norms. We have proposed a new mechanism to boost the robustness of blind watermarking scheme to cropping attacks.

References

[1] R. Ohbuchi, H. Masuda, and M. Aono, ”Watermarking Three-Dimensional Polygonal Models”, Proceedings of the ACM Multimedia, pp. 261-272, November 1997.

[2] F. Cayre, P. Rondao-Alface, F. Schmitt and B. Macq, H. Maitre, ”Application of Spectral Decomposition to Compression and Watermarking of 3D Triangle Mesh Geometry”, Signal Processing Volume 18, pp. 309319, April (2003).

[3] J. Wang, J. Feng and Y. Miao, ”A Robust Confirmable Watermarking Algorithm for 3D Mesh based on Manifold Harmonics Analysis”, The Visual Computer Volume 28, pp. 1049-1062, (2012).

[4] L. Chen, X. Kong, B. Weng, Z. Yao and R. Pan, ”A Novel Robust Mesh Watermarking Based on BNBW”, Advances in Signal Processing, pp.1-9, (2011).

[5] L. Jing, W. Yinghui, H. Wenjuan and L. Ye, ”A New Watermarking Method of 3D Mesh Model”, TELKOMNIKA Indonesian Journal of Electrical Engineering, pp.1610-1617, (2014).

[6] X. Rolland-Neviere, G. Doerr and P. Alliez, ”Triangle Surface Mesh Watermarking Based on a Constrained Optimization Framework”, IEEE Transactions on Information Forensics and Security, (2014).

[7] X. Mao, M. Shiba and A. Imamiya, ”Watermarking 3D Geometric Models through Triangle Subdivision”, Proceedings of the SPIE 4314: 253-260, (2001).

[8] F. Cayre and B. Macq B, ”Data hiding on 3-D triangle meshes”, IEEE Transactions on Signal Processing 51: 939-949, (2003).

[9] N. Werghi, N. Medimegh and S. Gazzah, ”Data Embedding of 3D Triangular Mesh Models using Ordered Ring Facets”, SSD, (2013).

[10] N. Werghi, M. Rahayem and J. Kjellander, ”An ordered topological representation of 3D triangular mesh facial surface: Concept and applications”, EURASIP Journal on Advances in Signal Processing, pp.144, (2012).

[11] J. W. Cho, R. Prost and H. Y. Jung, ”An oblivious watermarking for 3-D polygonal meshes using distribution of vertex norms”, IEEE Transactions on Signal Processing 55: 142-155, (2007).

[12] R. Hu, P. R. Alface and B. Macq, ”Constrained optimisation of 3d polygonal mesh watermarking by quadratic programming”, IEEE International Conference on Acoustics, Speech and Signal Processing, pp.1501-1504, (2009).

[13] M. Luo and A. G. Bors, ”Shape watermarking based on minimizing the quadric error metric”, IEEE International Conference on Shape Modeling and Applications, pp. 103-110, (2009).

[14] Y. Z. Zhan, Y. T. Li, X. Y. Wang and Y. Oian, ”A blind watermarking algorithm for 3D mesh models based on vertex curvature”, J Zhejiang Univ-Sci C (Comput Electron) 15: 351-362, (2014).

[15] N. Medimegh, S. Belaid and N. Werghi, ”A Survey of the 3D Triangular Mesh Watermarking Techniques”, International Journal of Multimedia, Volume: 1.1, (2015).

[16] B. Vallet and B. Levy, ”Spectral geometry processing with manifold harmonics”, Computer Graphics Forum, Vol. 27, Wiley Online Library, pp. 251260, 2008.

[17] H. H. Mohamed and S. Belaid, ”Algorithm BOSS (Bag-Of-Salient local Spectrums) for non-rigid and partial 3D object retrieval”, Neurocomputing, http://dx.doi.org/10.1016/j.neucom.2015.05.045.

[18] K. Gbal, J. A. Brentzen, H. Aans and R. Larsen, ”Shape analysis usingthe auto diffusion function”, Computer Graphics Forum, 2009, pp.14051413.

[19] J. A. Sethian, ”Level Set Methods and Fast Marching Methods Evolving Interfaces in Computational Geometry”, Fluid Mechanics, Computer Vision, and Materials Science. Cambridge, U.K.: Cambridge Univ., 1999.

[20] R. Kimmel and J. A. Sethian, Computing geodesic paths on manifolds, Proc. Nat. Acad. Sci.USA, vol. 95, pp. 84318435, 1998.

[21] Liu. Y-J, Xu. C-X and He. Y, Constructing Intrinsic Delaunay Triangulations from the Dual of Geodesic Voronoi Diagrams, arXiv:1505.05590[cs.CG].

[22] P. Cignoni, C. Rocchini and R. Scopigno, ”Metro: Measuring error on simplied surfaces”, Computer Graphic Forum 17: 167-174, (1998).