

A Robust Spectral Approach for Blind Watermarking of Manifold Surfaces

Yang Liu
Dept. of Computer Science
University of Texas at Dallas
Richardson, TX 75080, USA
yx1072100@utdallas.edu

Balakrishnan
Prabhakaran
Dept. of Computer Science
University of Texas at Dallas
Richardson, TX 75080, USA
praba@utdallas.edu

Xiaohu Guo
Dept. of Computer Science
University of Texas at Dallas
Richardson, TX 75080, USA
xguo@utdallas.edu

ABSTRACT

This paper proposes a robust, blind, and imperceptible spectral watermarking approach for manifold surfaces represented as triangle meshes. The basic idea is to transform the original mesh into frequency domain using the Fourier-Like *Manifold Harmonics Transform*. The manifold harmonics basis defined on arbitrary topology surfaces is an intrinsic property of the manifold surfaces, i.e., it is only determined by the surface metric and independent of their resolution and embedding. This property makes our watermarking scheme immune to uniform affine attack (rotation, scaling, and translation) and robust against noise-addition and mesh simplification attacks. The global manifold harmonics are computed using the finite element method combined with a band-by-band algorithm that can compute thousands of eigenvectors for large meshes with up to a million triangles. The watermark data is embedded by modifying the manifold harmonics descriptors magnitude in an imperceptible way. By using global spectral analysis, the detection of such watermarks does not require mesh registration or re-sampling, and analysis of the statistics of the manifold harmonics descriptors is exploited to devise an optimal blind detector. Experimental results show the imperceptibility of the watermark with low distortions, and its robustness against the most common attacks including the uniform affine transformations, random noise addition, mesh simplification, etc.

Categories and Subject Descriptors

K.6.5 [Management of Computing and Information Systems]: Security and Protection; I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling

General Terms

Security

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MM&Sec'08, September 22–23, 2008, Oxford, United Kingdom.
Copyright 2008 ACM 978-1-60558-058-6/08/09 ...\$5.00.

Keywords

3D Mesh, Spectral Watermark, Blind, Manifold Harmonics Transform

1. INTRODUCTION

Manifold surfaces representing 3D geometric models (e.g. triangle meshes) are widely used in CAD/CAM, video games, medical imaging and movies industries. Since it could be a time-consuming effort to generate and process these models, theft of these models would result in loss of time, money, and effort. Watermarking is a good way to protect the copyright of digital 3D models by embedding information into their surface representation [10, 6]. The presence of the watermark verifies the copyright.

According to Wang et al.'s survey [14], watermarking techniques for 3D models represented as triangle meshes can be classified into two main categories: Spatial and Spectral approaches, depending on whether the watermark is embedded in the spatial domain (modifying the geometry or the connectivity) or in the spectral domain (modifying some spectral coefficients). Depending on whether the original model should be available or not during watermark detection, the methods can be further characterized as non-blind or blind. Obviously, the availability of the original data makes non-blind detection much easier than blind detection. However, at the same time, it also limits the applicability of non-blind methods in real-world setups.

Spatial Techniques: The spatial description of a 3D mesh includes both geometry and connectivity information. Spatial techniques modify the spatial description to embed information. Mesh-based blind watermarking techniques are relatively vulnerable to connectivity attacks, such as re-meshing or mesh simplification [11]. Different methods have been developed for watermarking point-sampled surfaces [2, 3]. However, some of them are non-blind [9]. For blind schemes, the model registration is a difficult problem [14]. Some of the blind methods such as [16] try to find the center of gravity and use principal component analysis for watermarking. However, such methods are vulnerable to cropping as the center of gravity shifts and the watermarks are lost. Alface et al. [4] also localizes watermarks in features (e.g. head of the Stanford bunny model) of 3D models. The basic idea is to identify features of 3D models, and try to achieve robustness and imperceptibility per feature watermark. However, these techniques are comparatively less robust to global attacks as compared to local attacks.

Spectral Techniques: Most available spectral water-

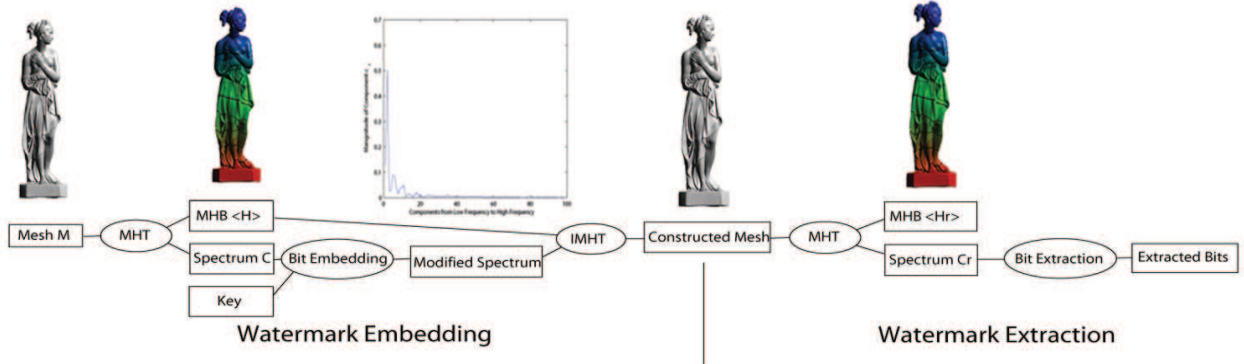


Figure 1: The pipeline of our watermarking scheme.

marking techniques have been focusing mainly on classical media data types like audios, images, and videos. Due to their regularly parameterized functional representations, most watermarking schemes are based on spread-spectrum methods with signal processing, i.e., the media data have to be transformed into a spectral domain, then the coefficients corresponding to the most perceptually salient basis functions will be modulated with watermarks to achieve robustness against possible attacks. Extending these spectral watermarking methods (for audios, images, and videos) to the newly emerged multimedia data type of 3D geometry models is difficult mainly because of the lack of basic manifold signal processing tools like regular parameterization and frequency analysis.

Spectral techniques, by converting meshes from spatial domain to spectral domain, are normally robust to global attacks like adding noises, simplification, and re-meshing. Spectral analysis is the key step in most spectral techniques. Recent spectral domain watermarking algorithms employed the spectral mesh analysis proposed by Karni and Gotsman [7]. Eigenvectors of the Laplace matrix to the input mesh, the Laplace basis functions, can span an ideal spectral space for robust watermarking, i.e., leading coefficients corresponding to smallest eigenvalues can be modulated with watermarks [8, 1]. This idea was generalized to watermark the point-based geometry in [5] where k -nearest neighbors have to be constructed to compose the Laplace matrices. These methods use the combinatorial graph Laplacian which does not take the geometric information into spectral decomposition, as compared to our geometry-aware discrete Laplacian used in this paper. [15] presented another spectral watermarking approach based on the orthogonalization of a small set of radial basis functions (RBFs) that can efficiently handle large meshes even with more than 10^6 vertices. However, all these existing spectral watermarking approaches for 3D models are non-blind, which requires surface registration in the watermark extraction stage.

Multiresolution analysis is closely related to spectral techniques. Wavelet is a common multiresolution analysis tool used in watermarking [12]. But according to [14], these multiresolution analysis approaches have either connectivity restrictions or robustness deficiencies, especially to connectivity-

changing attacks such as re-meshing, re-triangulation and simplification.

1.1 CONTRIBUTIONS

The objective of this research is to improve over existing techniques, and propose a blind and spectral watermarking approach using the geometry-aware discrete Laplacian basis [13] called *Manifold Harmonics*. Finite element method is employed to compute the manifold harmonics which are both geometry-aware and orthogonal on arbitrary 3D mesh domain. The global manifold harmonics are computed using a band-by-band algorithm that can compute thousands of eigenvectors for large meshes with up to a million triangles. The watermark data is embedded by modifying the magnitudes of manifold harmonics descriptors in an imperceptible way. Although the manifold harmonics transform is a Fourier-like transform, the computed orthogonal bases are intrinsic to the manifold domain. This means that the bases will not remain the same after watermark embedding since the shape has been slightly perturbed. This fact makes the watermark embedding and blind extraction a challenging task. We employ iterative embedding technique to overcome this difficulty. By using the global spectral analysis, the detection of such watermarks does not require mesh registration or re-sampling, and analysis of the statistics of the manifold harmonics descriptors is exploited to devise an optimal blind detector that achieves robustness against global attacks (noise addition, re-meshing, and simplifications), in addition to uniform affine transforms.

The watermarking scheme presented in this paper is *blind* in nature and satisfies the following properties:

Imperceptibility: As a spectral approach, this method is imperceptible due to “spread spectrum” principle [14], i.e., modification to original data will be spread over the whole mesh.

Multiple bits embedding: We divide the whole spectrum of the 3D mesh into multiple slots in frequency domain, and embed 1 bit of watermark in each slot. Thus multiple bits can be embedded after a wide spectrum being computed.

Robustness against attacks: Due to the intrinsic property of Laplace-Beltrami operator and manifold harmonics transform, this approach is immune against uniform affine trans-

formation without employing model registration (synchronization) which is a difficult problem [14].

2. SCHEME DESIGN

Our watermarking scheme employs the Manifold Harmonics Transform (MHT) to transform original mesh into spectral domain, after which we modify spectral coefficients to embed the sequence of watermark bits. Figure 1 shows the whole pipeline of our watermark embedding and extraction scheme. In the following subsection, we briefly review the concept of Manifold Harmonics, and how to use it to convert the functions defined on a manifold surface from spatial domain to frequency domain, and vice-versa. More detailed introduction can be found in [13, 17].

2.1 Manifold Harmonics Transform

In Euclidean domain \mathbb{R}^n Laplace operator is defined as the divergence of the gradient:

$$\Delta = \text{div grad} = \nabla \cdot \nabla = \sum_i \frac{\partial^2}{\partial x_i^2}. \quad (1)$$

By using exterior calculus (EC), the definition of the Laplacian can be generalized to functions defined over a manifold \mathcal{M} with metric g , and is then called the Laplace-Beltrami operator:

$$\Delta = \text{div grad} = \sum_i \frac{1}{\sqrt{|g|}} \frac{\partial}{\partial x_i} (\sqrt{|g|} \sum_j g^{ij} \frac{\partial}{\partial x_j}), \quad (2)$$

where $|g|$ denotes the determinant of g and g^{ij} denote the components of the inverse of the metric tensor g . Hence we can define the eigenfunctions and eigenvalues of the Laplacian on a manifold surface \mathcal{M} as all the pairs (H^k, λ_k) that satisfy :

$$-\Delta H^k = \lambda_k H^k. \quad (3)$$

The eigenfunctions of the continuous Laplace-Beltrami operator give orthogonal bases for the space of functions (e.g. geometric coordinates) defined on the manifold surface. Smaller eigenvalues of the spectrum are correlated to coarser features of the manifold while higher eigenvalues represent finer structures.

Finite Element Method (FEM) is employed to solve the eigenfunctions numerically. To set up the FEM, we need to define the *basis functions* used to express the solutions, and *test functions* onto which the equation (3) will be projected. For both basis and test functions, we choose the same set of “hat” functions Φ^i ($i = 1 \dots n$) that are associated with the n vertices, and are piecewise-linear on their 1-ring triangle faces. Then, we can express the eigenfunctions in the form: $H^k = \sum_{i=1}^n H_i^k \Phi^i$ and solve equation (3) by projection on the Φ^i ’s:

$$\forall j \in \{1, \dots, n\}, \quad \langle -\Delta H^k, \Phi^j \rangle = \lambda_k \langle H^k, \Phi^j \rangle, \quad (4)$$

or in matrix form:

$$-Qh^k = \lambda B h^k, \quad (5)$$

where $Q_{i,j} = \langle \Delta \Phi^i, \Phi^j \rangle$, $B_{i,j} = \langle \Phi^i, \Phi^j \rangle$, and h^k is the vector $[H_1^k, \dots, H_n^k]^T$. The matrix B is called *mass matrix*, and it can be replaced with a diagonal matrix D called the *lumped mass matrix* which is defined by $D_{i,i} = \sum_j B_{i,j}$.

The solution to this eigenproblem yields a series of eigenpairs (H^k, λ_k) called the Manifold Harmonics Basis (MHB).

The bases are orthogonal, i.e. the functional inner product $\langle H^i, H^j \rangle = 0$ if $i \neq j$. We also ensure that the MHB is orthonormal, by dividing each basis vector H^k by its functional norm $\langle H^k, H^k \rangle$. By using Arnoldi method [13], it is possible to compute eigenvectors band-by-band utilizing the shift-invert spectral transform. In our watermarking scheme, only eigenvectors for low-frequency components are computed, making our approach applicable to large models.

The geometry x (resp. y, z) of the triangulated surface \mathcal{M} can be considered as a piecewise linear function defined as a linear combination of the basis function $\Phi^i : x = \sum_{i=1}^n x_i \Phi^i$ where x_i denotes the x coordinate at vertex i .

Computing the Manifold Harmonics Transform (MHT) of the function x means converting x from the “hat” function basis (Φ^i in geometric domain) into the MHB (H^k in spectral domain). The MHT of x is the manifold harmonics descriptor which is a vector $[\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_m]$ given by:

$$\tilde{x}_k = \langle x, H^k \rangle = \mathbf{x}^T D h^k = \sum_{i=1}^n x_i D_{i,i} H_i^k, \quad (6)$$

where \mathbf{x} denotes the vector $[x_1, x_2, \dots, x_n]^T$.

The Inverse Manifold Harmonics Transform (IMHT) maps the descriptor from frequency domain into geometric domain. The reconstructed coordinate at a vertex is given by:

$$x_i = \sum_{k=1}^m \tilde{x}_k H_i^k. \quad (7)$$

Figure 2 shows the manifold harmonics basis for the Kitten model with different resolutions. Their spectral bases are extremely similar although the mesh resolution and connectivity are different.

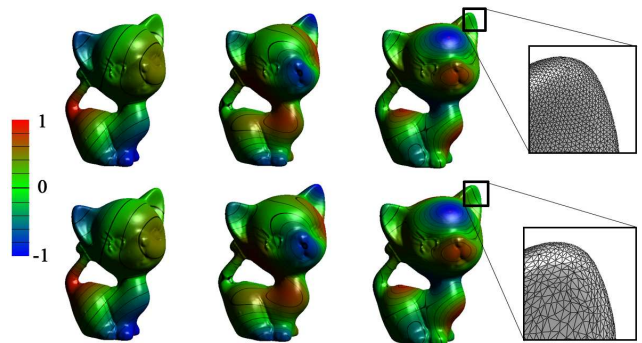


Figure 2: The Manifold Harmonics Bases H^6 (1st column), H^{17} (2nd column), H^{26} (3rd column) of the Kitten mesh with different resolutions: 268,000 faces (1st row) and 90,000 faces (2nd row). The spectral bases are extremely similar between different resolutions.

2.2 Bit Embedding

Given a 3D mesh M , our watermarking scheme transforms it into spectral domain using Manifold Harmonics Transform (equation (6)), and modifies the mesh by embedding watermark bits into the spectral coefficients. Then a new mesh

M_r will be constructed based on the modified spectral coefficients using the Inverse Manifold Harmonics Transform (equation (7)).

2.2.1 Assumption on MHB Similarity

Suppose the original mesh M has MHB $\{H^i | i = 1 \dots m\}$. The corresponding spectrum of M is $\{(\tilde{x}_i, \tilde{y}_i, \tilde{z}_i) | i = 1 \dots m\}$. The objective of this approach is to embed a sequence of bit $k[i]$ into the spectrum. Here the bit sequence $k[i]$ is called the **key**. By slightly modifying $(\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)$ to embed watermark bits, we have $(\tilde{x}'_i, \tilde{y}'_i, \tilde{z}'_i)$. With modified spectrum $\{(\tilde{x}'_i, \tilde{y}'_i, \tilde{z}'_i)\}$ and $\{H^i\}$, we can reconstruct a watermarked mesh M_r .

However, the MHB of M_r , $\{H_r^i\}$, will be slightly different from $\{H^i\}$, because the manifold surface of M_r is slightly perturbed from the original mesh M . The new set of MHB $\{H_r^i\}$ will be identical to the original $\{H^i\}$ only when the mesh undergoes isometric transformation, which can not hold for watermark embedding process. Since we are developing a blind watermarking scheme, the new set of basis $\{H_r^i\}$ will be used in the watermark extraction stage where no information about the original mesh is available.

Fortunately, the distortion introduced to the meshes using our watermarking scheme is very small (see Section 3.2 for more detailed discussion). So an important assumption in this paper is that $\{H_r^i\}$ and $\{H^i\}$ should be very similar. According to this assumption, the modified spectrum of M should be similar to the spectrum of M_r . That is, $(\tilde{x}'_i, \tilde{y}'_i, \tilde{z}'_i)$ should be similar to $(\tilde{x}_{r_i}, \tilde{y}_{r_i}, \tilde{z}_{r_i})$ which is the spectrum of M_r .

Given the meshes M and M_r , we use the following metric to compute the similarity between their corresponding MHB:

$$\text{similarity} = \frac{\langle H_r^i, H^i \rangle}{\|H_r^i\| \cdot \|H^i\|} \in [0, 1] \quad (8)$$

where $\langle H_r^i, H^i \rangle$ stands for the functional inner product of between the two bases H_r^i and H^i . $\|H^i\|$ stands for the L_2 norm of the basis function H^i .

Figure 3 shows the similarity of the corresponding MHB between the Iphigenia mesh and its modified mesh in which the 50th component is embedded with 1 bit of watermark. From this figure, we can see that the two bases H_r^i and H^i are very similar (with value close to 1) for most of the frequency components.

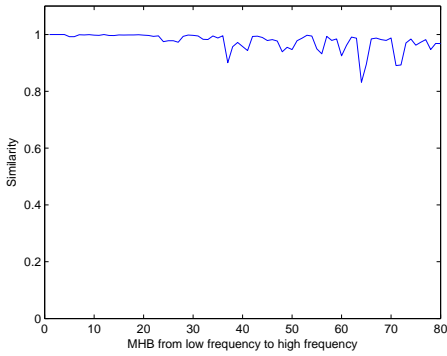


Figure 3: Similarity between the bases of the Iphigenia mesh and its modified mesh.

2.2.2 Rotational Invariant

To make our approach immune to rotational attack, we employ the spectral coefficient magnitude which is defined as:

$$c_i = \sqrt{\tilde{x}_i^2 + \tilde{y}_i^2 + \tilde{z}_i^2}. \quad (9)$$

When a rotation R is applied to the model mesh in spatial domain, according to equation (6) the spectral coefficients $(\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)$ will be rotated as well. This makes c_i invariant to rotation. By embedding and extracting watermark with the spectral coefficient magnitudes, our method should be robust to rotational attack.

2.2.3 Alternative Ways of Bit Embedding

To achieve the contradictory goals of both minimizing the distortion introduced by watermark embedding and making the watermarking scheme robust to noises, only middle frequency components are used to embed watermark. All middle frequency components are divided into slots $S_i = \{c_{i,j}\}$ with the same size (we use 10 as the size for all of our experiments). Each slot S_i is used to embed 1 bit of watermark. We used the following 2 alternative ways for bit embedding.

Aggressive Bit Embedding: In each slot $S_i = \{c_{i,j}\}$, only 1 frequency component c_{i,s_i} is selected to be modified. The relative position of the selected component c_{i,s_i} in each slot is not fixed, and this information is only known to the embedding and extraction systems. Adjacent components in S_i are used to calculate average value a_i and deviation d_i of the spectral coefficient magnitudes in S_i .

To embed a specified bit $k[i]$ in each slot S_i , we modify c_{i,s_i} as follows:

$$\begin{cases} c'_{i,s_i} = c_{i,s_i}, & \text{if } k[i] = 1, c_{i,s_i} > a_i + s \cdot d_i \\ c'_{i,s_i} = a_i + s \cdot d_i, & \text{if } k[i] = 1, c_{i,s_i} < a_i + s \cdot d_i \\ c'_{i,s_i} = a_i - s \cdot d_i, & \text{if } k[i] = 0, c_{i,s_i} > a_i - s \cdot d_i \\ c'_{i,s_i} = c_{i,s_i}, & \text{if } k[i] = 0, c_{i,s_i} < a_i - s \cdot d_i \end{cases} \quad (10)$$

Here s stands for the turbulence factor. Small factor s may only distort the original mesh slightly. However, it can cause the watermarking scheme vulnerable to noises as well.

Non-Aggressive Bit Embedding: Similar to Aggressive Bit Embedding, Non-Aggressive Bit Embedding also modify c_{i,s_i} to embed 1 bit into each slot. The only difference is on the method to modify c_{i,s_i} :

$$\begin{cases} c'_{i,s_i} = c_{i,s_i}, & \text{if } k[i] = 1, c_{i,s_i} > a_i \\ c'_{i,s_i} = a_i + s \cdot d_i, & \text{if } k[i] = 1, c_{i,s_i} < a_i \\ c'_{i,s_i} = a_i - s \cdot d_i, & \text{if } k[i] = 0, c_{i,s_i} > a_i \\ c'_{i,s_i} = c_{i,s_i}, & \text{if } k[i] = 0, c_{i,s_i} < a_i \end{cases} \quad (11)$$

2.3 Iterative Embedding

According to the assumption stated in Section 2.2.1, the magnitude coefficients $\{c_{r_i}\}$ of reconstructed mesh M_r should be similar to modified magnitude coefficients $\{c'_i\}$ of mesh M . Figure 4 shows the magnitude coefficient spectrum of the original Iphigenia mesh and its modified mesh. In this example the 50th component in the original mesh is doubled. That is, $c'_{50} = 2c_{50}$. We can see that the spectrum of the reconstructed mesh is similar to the spectrum of the original mesh. Figure 5 shows the visual appearances of the two Iphigenia meshes together with the color-coded geometric distortions (to be discussed in Section 3.2.2).

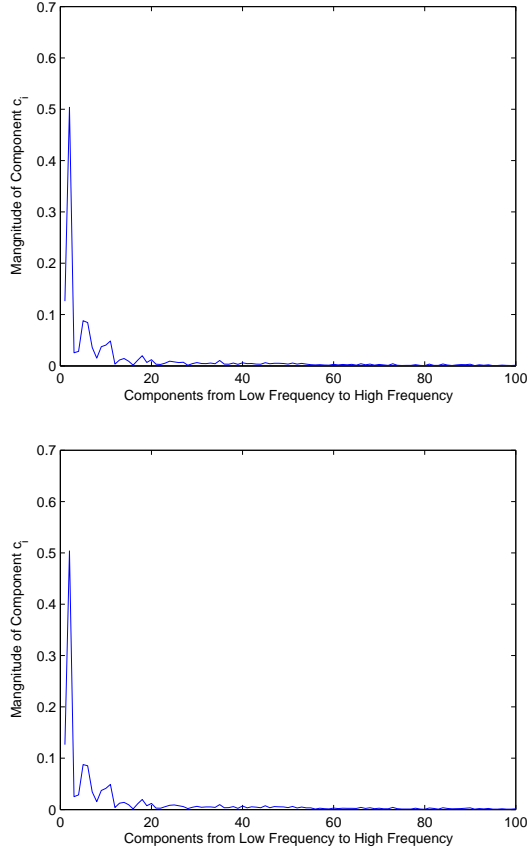


Figure 4: The spectrum of Iphigenia mesh: original (top) and modified (bottom).

However, there are still some differences between the two spectrum of meshes before and after watermarking. During watermark extraction, the reconstructed mesh M_r is projected to the MHB $\{H_r^i\}$ which is different from $\{H^i\}$ of the original mesh M . Then in the spectrum of reconstructed mesh, $c_{r_i} = c_i$ may not hold even if $c'_i = c_i$ holds. Figure 6 shows the difference between $\{c_{r_i}\}$ and $\{c_i\}$ for the Igea mesh. The visual appearances of the meshes are shown in Figure 7. From this example, we can see that although the spectrum of M_r is very similar to the spectrum of original mesh M , there are still some numerical differences. In fact the changes of spectrum made by a specific watermarking purpose as well. For the 50th component we modified, although $c_{r_{50}} > c_{50}$ holds, $c_{r_{50}} = c'_{50} = 2c_{50}$ does not hold. This makes the watermark embedding unreliable for extraction purpose.

To make watermark embedding more reliable, we employ an iterative embedding approach. That is, after getting the re-constructed M_r , we use M_r as original mesh and perform the embedding again. Thus we take the following steps:

- (1) Transform the mesh M into spectral components $\{c_i\}$;
- (2) Split the middle frequency components into adjacent slots, modify 1 component for each slot to embed 1 bit of watermark;
- (3) Re-construct mesh M_r ;



Figure 5: The Iphigenia model (left), the modified Iphigenia model (middle), and the color-coded geometric distortion (right).

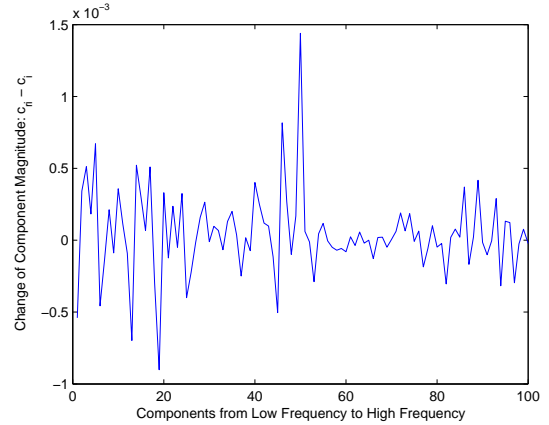


Figure 6: The spectrum difference between the Igea mesh and its modified mesh: $c_{r_i} - c_i$.

- (4) Extract embedded bits from mesh M_r . If the number of correct bits exceeds the extraction threshold then output M_r , otherwise replace M with M_r and go to step 1.

Note that higher extraction threshold may require more iterations, and may also introduce more distortions onto the mesh M_r . Figure 8 shows the evolution of c_{50} and the average value of the corresponding slot (from the 45th to 54th frequency components) for the Igea model. In this slot the bit 1 is embedded. That is we modify c_{50} to make it larger than the average value. As shown in the figure, although this is not achieved in one iteration, the difference between c_{50} and the average value keeps decreasing until the goal is achieved.

2.4 Watermark Extraction

By converting the given meshes into spectral domain, the spectral coefficient magnitude c_i , the average value a_i , and the deviation d_i are calculated for each group of frequency components. By comparing c_{i,s_i} and a_i , we can extract one bit w_i from each group G_i :

$$\begin{cases} w_i = 1, & \text{if } c_{i,s_i} > a_i \\ w_i = 0, & \text{if } c_{i,s_i} < a_i \end{cases} \quad (12)$$

By comparing the extracted bits $\{w_i\}$ with the original key $\{k[i]\}$, we can assert whether a given mesh is embedded with watermarks or not.



Figure 7: The Igea model (left), the modified Igea model (middle), and the color-coded geometric distortion (right).

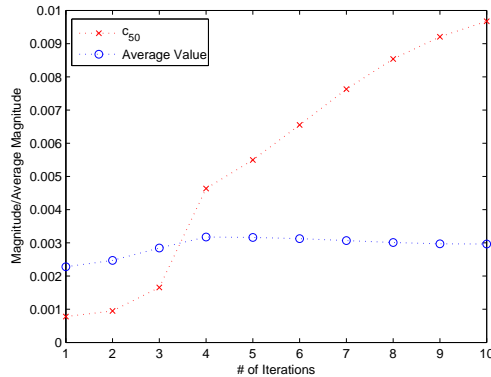


Figure 8: Evolution of component c_{50} and the average value for the Igea model during the iteration process.

3. PERFORMANCE ANALYSIS

Our watermarking scheme was developed using Microsoft Visual C++ 2005. We conducted our experiments on a Windows XP platform with 2.66GHz CPU and 2GB RAM.

3.1 Efficiency

To improve efficiency, we employ Arnoldi method implemented in ARPACK library to solve the eigenvalue problem band-by-band in the Manifold Harmonic Transform [13]. By using Arnoldi method, it is possible to solve only the necessary eigenvalues and eigenvectors, which can significantly reduce computational effort. In our experiments, it takes less than 1 minute to calculate the Manifold Harmonic Basis for a mesh with 20,000 vertices. The other operations like watermark embedding and extraction only take several seconds.

3.2 Distortion

Imperceptibility is the key feature of our watermarking approach. To achieve better imperceptibility of embedding, we try several alternative ways within our scheme to find the best approach with least distortion. As mentioned in Section 2.2.3, we have two alternative ways to embed bits into the spectrum. Another degree of freedom depends on how we split the middle frequency components into slots.

3.2.1 Slots Division

Besides the two alternative ways of bit embedding men-

tioned in Section 2.2.3, we also test our approach with different size of slots. In our experiments we try to embed a key of 5 bits into the meshes, and choose the 21st to 70th frequency components to embed these 5 bits.

Normal Slots: In the Normal Slots mode, each slot contains 10 frequency components $\{c_{i,1} \dots c_{i,10}\}$. The components from $c_{i,1}$ to $c_{i,9}$ except $c_{i,5}$ are used to calculate average value and deviation, while $c_{i,5}$ is selected to be modified. $c_{i,10}$ is not referred. Thus $c_{i,5}$ is in the center of referred components in each slot, and can be embedded with 1 bit of the key.

Interleaved Narrow Slots: Interleaved Narrow Slots is similar to Normal Slots. The difference is that in each slot only $c_{i,4}$ and $c_{i,6}$ are used to calculate average value and deviation. That is, there will be 7 un-referred components between the 3 referred components in adjacent slots.

With the two alternative ways of bit embedding and the two alternative ways of slots division, now we have 4 combined ways to test our watermarking scheme.

3.2.2 Distortion Measurement

In order to measure the distortion caused by the watermark embedding, a metric is required to capture both the geometric difference and the visual difference between the original mesh and the watermarked mesh. The simplest measure is just the geometric distance between corresponding vertices in both models.

$$GeomDiff(M_1, M_2) = \frac{1}{n} \sum_{v_1 \in M_1, v_2 \in M_2} \|v_1 - v_2\|. \quad (13)$$

While this does give some indication of geometric closeness, it does not capture the more subtle visual properties the human eye appreciates, such as curvature change. Geometric Laplacian is a good approximation for the mean curvature of the surface mesh, and can be defined as:

$$GL(v_i) = v_i - \frac{\sum_{j \in N(i)} l_{ij}^{-1} v_j}{\sum_{j \in N(i)} l_{ij}^{-1}}, \quad (14)$$

where $N(i)$ is the set of indices of the neighbors of vertex i , and l_{ij} is the geometric distance between vertices i and j . Thus the curvature difference between two mesh M_1 and M_2 is defined as:

$$CurvDiff(M_1, M_2) = \frac{1}{n} \sum_{v_1 \in M_1, v_2 \in M_2} \|GL(v_1) - GL(v_2)\|. \quad (15)$$

3.2.3 Experimental Data

Since there are 4 alternative ways to test our watermarking scheme, we can choose the way that introduce the least distortion into the original mesh. We use equation (13) and (15) to measure the geometric and curvature distortions introduced by the watermark embedding process.

Normal Slots with Aggressive Bit Embedding: Table 1 shows the number of iteration required by each model to embed all bits successfully, together with their geometric and curvature distortions. Figure 9 shows the color-coded distortion distribution on the Dinosaur model.

Normal Slots with Non-Aggressive Bit Embedding: As shown in Table 2, with Non-Aggressive Bit Embedding there is only little difference in the converging speed. Watermark-embedded models also look very similar to those with Ag-

Table 1: Iterations in Normal Slots with Aggressive Mode

Model	Iteration	Geom Diff	Curv Diff
BallJoint	2	0.234169%	0.233259%
Buddha	9	0.747521%	0.742569%
Dinosaur	3	0.843850%	0.841414%
Elephant	3	1.300157%	1.297724%
Gargoyle	3	1.080153%	1.073566%
Greek	2	0.505423%	0.502522%
Horse	3	0.894512%	0.892042%
Igea	2	0.476169%	0.475276%
Iphigenia	4	0.512932%	0.510703%
Neptune	3	0.792230%	0.791762%
Shell	4	1.209168%	1.207613%
Teeth	1	0.252154%	0.251556%
Topology	7	1.482054%	1.460097%

gressive Bit Embedding. Figure 10 shows the color-coded distortion distribution on the Gargoyle model.

Table 2: Iterations in Normal Slots with Non-Aggressive Mode

Model	Iteration	Geom Diff	Curv Diff
BallJoint	2	0.179201%	0.178477%
Buddha	3	0.477365%	0.474270%
Dinosaur	3	0.774479%	0.772246%
Elephant	3	1.298359%	1.295943%
Gargoyle	3	0.928533%	0.922703%
Greek	2	0.481065%	0.478384%
Horse	3	0.867700%	0.865367%
Igea	2	0.454682%	0.453806%
Iphigenia	3	0.476333%	0.474304%
Neptune	3	0.798724%	0.798294%
Shell	4	0.925626%	0.924383%
Teeth	2	0.255855%	0.255231%
Topology	2	1.110981%	1.098596%

Narrow Slots with Aggressive Bit Embedding: As shown in Table 3, with Interleaved Narrow Slots, models tend to require more iterations to embed watermark bits successfully. Figure 11 shows the color-coded distortion distribution on the Greek model.

Narrow Slots with Non-Aggressive Bit Embedding: Table 4 shows the number of iteration required by each model to embed all bits using Interleaved Narrow Slots with Non-Aggressive Mode, together with their geometric and curvature distortions. Figure 12 shows the color-coded distortion distribution on the Shell model.

3.3 Robustness to Attacks

3.3.1 Uniform Affine Transform Attack

Rotation Attack: When we apply a rotation R to the model in spatial domain, the descriptors $(\tilde{x}_i, \tilde{y}_i, \tilde{z}_i)$ in the frequency domain will be rotated by R as well, making the length of the descriptor vectors invariant under rotations. Since we use the magnitude $\{c_i\}$ of the descriptor vectors as spectrum to embed digital watermarks, our method is naturally immune to rotational attacks.

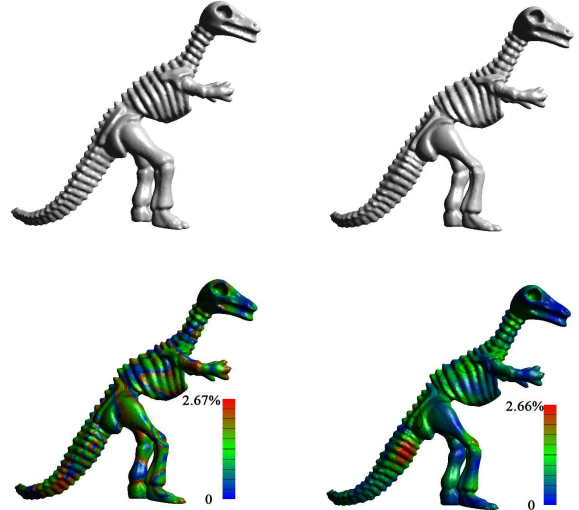


Figure 9: The original (top-left) and watermarked (top-right) Dinosaur model, with the color-coded geometric distortion (bottom-left) and curvature distortion (bottom-right), using the Normal Slots with Aggressive Embedding mode.

Uniform Scaling Attack: Let \mathcal{M} be a compact 2-dimensional Riemannian manifold with the local parameterization $h : \mathbf{R}^2 \rightarrow \mathbf{R}^3$. The scaled manifold $\bar{\mathcal{M}} = a\mathcal{M}$ with the parameterization $\bar{h} = ah$ possesses the partial derivatives: $\partial_i \bar{h} = a\partial_i h$ ($i = 1, 2$), implying $\bar{g}^{ij} = \frac{1}{a^2}g^{ij}$ and $\sqrt{|\bar{g}|} = \frac{1}{a^2}\sqrt{|g|}$. With u being a solution to

$$\Delta_h u = \frac{1}{\sqrt{|g|}} \sum_{i,j} \partial_i (g^{ij} \sqrt{|g|} \partial_j u) = -\lambda u,$$

we can find u as a solution to

$$\Delta_{\bar{h}} u = \frac{1}{\sqrt{|\bar{g}|}} \sum_{i,j} \partial_i (\bar{g}^{ij} \sqrt{|\bar{g}|} \partial_j u) = -\frac{1}{a^2} \lambda u.$$

This implies that the eigenvalues $\bar{\lambda}_i = \frac{1}{a^2} \lambda_i$. Since we normalize each basis vector $\langle \bar{H}^k, \bar{H}^k \rangle = \langle H^k, H^k \rangle = 1$, with the parameterization relationship $\bar{h} = ah$, we can see that the basis vectors have the relationship: $\bar{H}^k = \frac{1}{a} H^k$.

Considering equation (6) and (9), we have:

$$\bar{c}_i = a^2 c_i.$$

By applying the extraction rule of equation (12), there is:

$$\bar{w}_i = w_i$$

This implies that our approach is immune to the uniform scaling attacks.

Translation Attack: Note that the first manifold harmonics basis is a constant field function $H^1 = c$ corresponding to eigenvalue $\lambda_1 = 0$. So the first basis H^1 will be able to capture all the translational effects of the underlying model after MHT. Because we choose NOT to embed watermarks into the first descriptor $(\tilde{x}_1, \tilde{y}_1, \tilde{z}_1)$, the watermarking scheme is naturally immune to translational attacks, without any heuristic pre-processing based on computing the centers of gravity of the 3D models.

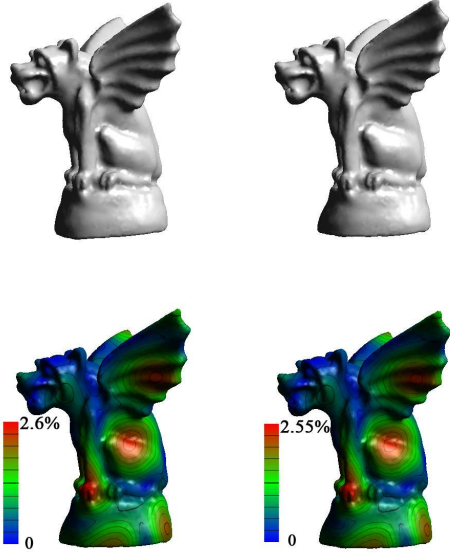


Figure 10: The original (top-left) and watermarked (top-right) Gargoyle model, with the color-coded geometric distortion (bottom-left) and curvature distortion (bottom-right), using the Normal Slots with Non-Aggressive Embedding mode.

3.3.2 Mesh Simplification Attack

Mesh simplification has been commonly used in geometric processing tasks. Because this kind of attack changes both the number of vertices and the connectivity between them (i.e. re-meshing), it is hard for spatial watermarking scheme to withstand the mesh simplification attacks.

Experiment results show that our approach is robust against simplification attacks. Figure 13 shows the watermarked Igea mesh and the mesh after simplification. All bits are extracted from the attacked mesh successfully.

3.3.3 Adding-Noise Attack

Additive White Gaussian Noise (AWGN) is the most common noise. We use the following formula to compute the position of vertices after adding the AWGN noise:

$$\mathbf{v}'_k = \mathbf{v}_k + s_{noise} \cdot AWGN(k) \cdot \bar{e} \cdot normal(\mathbf{v}_k),$$

where $AWGN(k)$ stands for the Additive White Gaussian Noise with average energy of one, \bar{e} stands for the average length of edges, $normal(\mathbf{v}_k)$ stands for the normal vector of vertex \mathbf{v}_k , and s_{noise} stands for the noise control factor. Figure 14 shows the watermarked Elephant mesh after the adding-noise attack with $s_{noise} = 0.5$. Figure 15 shows the watermarked Topology mesh after the adding-noise attack with $s_{noise} = 0.7$. In both cases all of the key bits are extracted successfully.

3.3.4 Smoothing Attack

Smoothing attack is performed by using the following equation for IMHT such that the high frequency components are

Table 3: Iterations in Narrow Slots with Aggressive Mode

Model	Iteration	Geom Diff	Curv Diff
BallJoint	7	0.304506%	0.303407%
Buddha	7	0.743958%	0.739356%
Dinosaur	4	0.776946%	0.774803%
Elephant	3	1.111846%	1.109944%
Gargoyle	8	2.024484%	2.012315%
Greek	3	0.548618%	0.545405%
Horse	8	1.241718%	1.238079%
Igea	4	0.317021%	0.316497%
Iphigenia	4	0.752914%	0.749533%
Neptune	3	0.585220%	0.584601%
Shell	4	1.052538%	1.050436%
Teeth	1	0.293767%	0.293044%
Topology	16	2.466684%	2.415274%

Table 4: Iterations in Narrow Slots with Non-Aggressive Mode

Model	Iteration	Geom Diff	Curv Diff
BallJoint	6	0.212013%	0.211199%
Buddha	4	0.532908%	0.529667%
Dinosaur	5	0.765400%	0.763228%
Elephant	5	1.114529%	1.112623%
Gargoyle	3	0.835289%	0.830061%
Greek	5	0.543010%	0.539735%
Horse	13	1.227770%	1.224172%
Igea	4	0.257239%	0.256824%
Iphigenia	8	0.589470%	0.586745%
Neptune	6	0.604990%	0.604353%
Shell	3	0.730253%	0.728965%
Teeth	2	0.311238%	0.310506%
Topology	11	1.707945%	1.671483%

reduced by 50 percent:

$$x_i = \sum_{k=1}^{200} \tilde{x}_k H_i^k + \sum_{k=201}^m \frac{1}{2} \tilde{x}_k H_i^k.$$

Figure 16 shows the watermarked Igea mesh after the smoothing attack. All key bits are extracted successfully.

4. CONCLUSIONS

In this paper, we propose a robust spectral watermarking approach for 3D mesh surfaces based on manifold harmonics. The manifold harmonics bases are not dependent on the mesh connectivity information, because the spectrum only depends on the gradient and divergence which are defined to be dependent only on the Riemannian structure of the manifold surfaces. Our watermarking scheme is naturally immune to uniform affine transformations without any heuristic computation of the center of gravity and principle component analysis, and is robust against mesh simplification, noise addition, and smoothing attacks. Through our extensive experiments on watermark embedding, extraction, and attacks, we demonstrate the improved performance of our blind spectral watermarking scheme. In the future, we will integrate our current spectral scheme with the feature-based mesh segmentation algorithms [4],

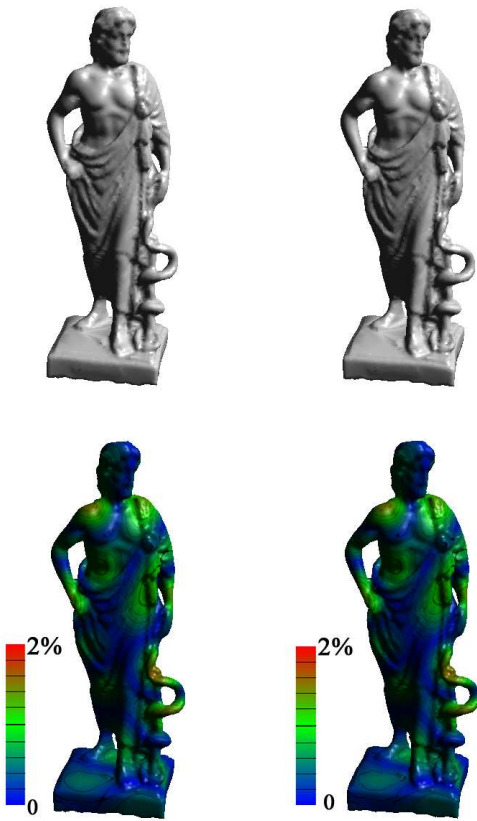


Figure 11: The original (top-left) and watermarked (top-right) Greek model, with the color-coded geometric distortion (bottom-left) and curvature distortion (bottom-right), using the Narrow Slots with Aggressive Embedding mode.

by investigating cropping-invariant mesh segmentation algorithms and the manifold harmonics transform for open surface patches, to make our blind watermarking framework robust against cropping attacks which may induce severe losses of shape information.

5. ACKNOWLEDGEMENTS

Y. Liu and X. Guo are partially supported by National Science Foundation under Grant No. 0727098. B. Prabhakaran is supported in part by US Army Research Office grant 48645-MA and National Science Foundation under Grant No. 0237954.

6. REFERENCES

- [1] E. E. Abdallah, A. B. Hamza, and P. Bhattacharya. Spectral graph-theoretic approach to 3D mesh watermarking. In *Proceedings of Graphics Interface*, pages 327–334, 2007.
- [2] P. Agarwal and B. Prabhakaran. Robust blind watermarking mechanism for point sampled geometry. In *Proceedings of the 9th ACM Multimedia and Security Workshop*, pages 175–186, 2007.
- [3] P. Agarwal and B. Prabhakaran. Robust blind watermarking of point sampled geometry. *IEEE*

Transactions on Information Forensics and Security, to appear, 2008.

- [4] P. R. Alface, B. Macq, and F. Cayre. Blind and robust watermarking of 3D models: How to withstand the cropping attack? In *Proceedings of the IEEE International Conference on Image Processing*, volume 5, pages 465–468, 2007.
- [5] D. Cotting, T. Weyrich, M. Pauly, and M. Gross. Robust watermarking of point-sampled geometry. In *Proceedings of the Shape Modeling International*, pages 233–242, 2004.
- [6] I. Cox, M. Miller, J. Bloom, and M. Miller. *Digital Watermarking: Principles & Practice*. Morgan Kaufmann, 2001.
- [7] Z. Karni and C. Gotsman. Spectral compression of mesh geometry. In *Proceedings of SIGGRAPH '00*, pages 279–286, 2000.
- [8] R. Ohbuchi, A. Mukaiyama, and S. Takahashi. A frequency-domain approach to watermarking 3D shapes. *Computer Graphics Forum*, 21(3):373–382, 2002.
- [9] R. Ohbuchi, A. Mukaiyama, and S. Takahashi. Watermarking a 3D shape model defined as a point set. *Proceedings of the International Conference on Cyberworlds*, pages 392–399, 2004.
- [10] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn. Information hiding – A survey. *Proceedings of the IEEE*, 87(7):1062–1078, 1999.
- [11] E. Praun, H. Hoppe, and A. Finkelstein. Robust mesh watermarking. In *Proceedings of SIGGRAPH'99*, pages 49–56, 1999.
- [12] F. Ucheddu, M. Corsini, and M. Barni. Wavelet-based blind watermarking of 3d models. In *MM&Sec '04: Proceedings of the 2004 workshop on Multimedia and security*, pages 143–154, New York, NY, USA, 2004. ACM.
- [13] B. Vallet and B. Lévy. Spectral geometry processing with manifold harmonics. In *Proceedings of Eurographics*, volume 27, pages 251–260, 2008.
- [14] K. Wang, G. Lavouè, F. Denis, and A. Baskurt. Three-dimensional meshes watermarking: Review and attack-centric investigation. In *Proceedings of the International Workshop on Information Hiding*, pages 50–64, 2007.
- [15] J. Wu and L. Kobbelt. Efficient spectral watermarking of large meshes with orthogonal basis functions. *The Visual Computer*, 21:848–857, 2005.
- [16] S. Zafeiriou, A. Tefas, and I. Pitas. Blind robust watermarking schemes for copyright protection of 3D mesh objects. *IEEE Transactions on Visualization and Computer Graphics*, 11(5):596–607, 2005.
- [17] H. Zhang, O. van Kaick, and R. Dyer. Spectral methods for mesh processing and analysis. In *Proceedings of Eurographics State-of-the-art Report*, pages 1–22, 2007.

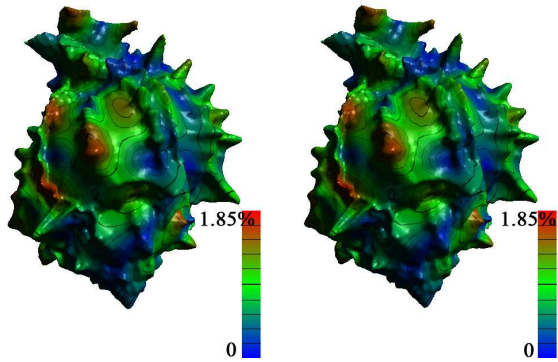
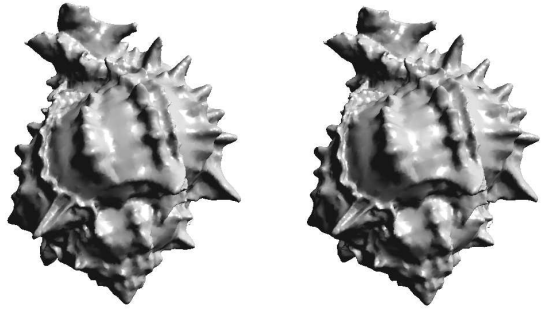


Figure 12: The original (top-left) and watermarked (top-right) Shell model, with the geometric distortion (bottom-left) and curvature distortion (bottom-right), using the Narrow Slots with Non-Aggressive Embedding mode.

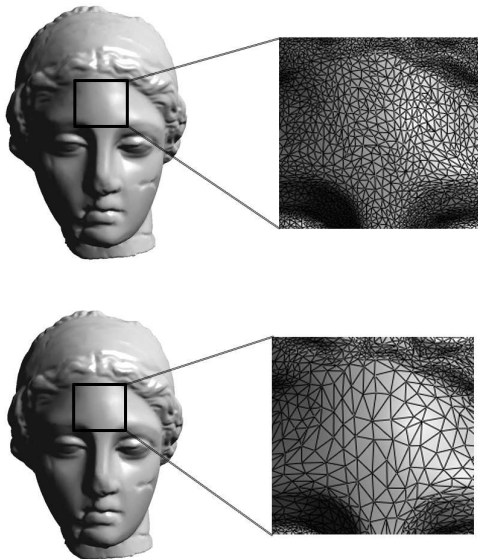


Figure 13: Watermarked Igea mesh before (top, with 100k faces) and after simplification (bottom, with 50k faces)



Figure 14: The original (left), watermarked (middle), and noise-added (right, with $s_{noise} = 0.5$) mesh of Elephant model, using the Normal Slots with Aggressive Embedding mode.

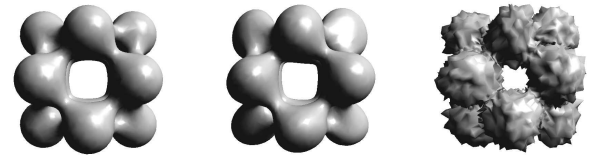


Figure 15: The original (left), watermarked (middle), and noise-added (right, with $s_{noise} = 0.7$) mesh of Topology model, using the Narrow Slots with Aggressive Embedding mode.



Figure 16: The original (left), watermarked (middle), and smoothed (right) mesh of Igea model, using the Normal Slots with Non-Aggressive Embedding mode.