

Numéro d'ordre : 2009-ISAL-0082 Année 2009

Institut National des Sciences Appliquées de Lyon
Laboratoire d'InfoRmatique en Image et Systèmes d'information
École Doctorale Informatique et Mathématiques de Lyon

Thèse de l'Université de Lyon

Présentée en vue d'obtenir le grade de Docteur,
spécialité Informatique

par

Kai Wang

Quantization-Based Blind Watermarking of Three-Dimensional Meshes

---

Thèse soutenue le 6 novembre 2009 devant le jury composé de :

M.	Mauro Barni	Professeur, Università di Siena	Rapporteur
M.	Bruno Lévy	Directeur de Recherche, INRIA Nancy Grand Est	Rapporteur
M.	Benoît Macq	Professeur, Université Catholique de Louvain	Rapporteur
M.	Jean-Marc Chassery	Directeur de Recherche CNRS, GIPSA-lab Grenoble	Examinateur
Mme.	Caroline Fontaine	Chargée de Recherche CNRS, IRISA Rennes	Examinateur
M.	Atilla Baskurt	Professeur, INSA Lyon	Directeur
Mme.	Florence Denis	Maître de Conférences, Université Lyon 1	Co-encadrant
M.	Guillaume Lavoué	Maître de Conférences, INSA Lyon	Co-encadrant

Laboratoire d'InfoRmatique en Image et Systèmes d'information
UMR 5205 CNRS - INSA de Lyon - Bât. Jules Verne
69621 Villeurbanne cedex - France
Tel: +33 (0)4 72 43 60 97 - Fax: +33 (0)4 72 43 71 17

To my grandfather.

献给我的外祖父。

Acknowledgments

First of all, I would like to express my gratitudes to my three thesis advisors for their consistent encouragements and support (on both scientific research and personal life) during the last three years. I would like to thank Pr. Atilla Baskurt for proposing to me a very interesting Ph.D. subject which crosses several different exciting research domains such as signal processing, computer graphics and shape analysis. I am very grateful to Dr. Florence Denis for her insightful suggestions and comments all along this thesis work. Finally, my sincere thanks go to Dr. Guillaume Lavoué for too many things: he is always ready to answer my questions that are expressed in poor French; he is so quick and efficient in correcting my papers so that I can do my work at my rhythm; and he is also the cautioner of my apartment lease...

I am very thankful to Pr. Mauro Barni, Dr. Bruno Lévy and Pr. Benoît Macq, for taking their precious time to review my thesis manuscript. I am also very grateful to Pr. Jean-Marc Chassery and Dr. Caroline Fontaine for having accepted to be examiners of my thesis defence.

I would like to thank Dr. Adrian G. Bors and Mr. Ming Luo from the University of York in England, for the fruitful discussions with them. I am also thankful to Peng Ren and Le Dong for their kind help during my research visit at York.

I am grateful to Dr. Bruno Lévy and Dr. Bruno Vallet for answering my questions on manifold harmonics and for making the relevant code freely available on-line.

I want to thank all the members of the M2DisCo and Imagine teams at LIRIS. Particularly, my thanks go to Dr. Céline Roudet and Dr. David Coeurjolly for their help on the coding and testing of the algorithms proposed in this thesis, to Pr. Florent Dupont for his financial support and also for the discussions with him which helped to improve the quality of my work, and to Djamel, Çağatay, Émilie, Yi and Phuong for the daily conversations as colleagues having shared the same office.

I thank Pr. Yuehu Liu, my Master thesis advisor at Xi'an Jiaotong University, for having persuaded me to do a Ph.D..

I thank my friends and my family members (especially my grandparents, my parents, my parents-in-law and my wife) for their unconditional support all along the years.

Finally, I would like to dedicate this thesis to my grandfather Mr. Xixian Zhou, a great and respectable person whom I lost during the writing of this manuscript.

Abstract

With the increasing use of three-dimensional (3-D) models in various practical applications, more and more attention has been paid on the research of digital watermarking techniques for 3-D polygonal meshes. In this thesis, we first provide a comprehensive survey on the state of the art in 3-D mesh watermarking, with an original attack-centric investigation. Then, we make use of the scalar Costa quantization scheme to construct a number of effective blind mesh watermarking schemes. We successfully embed multi-bit quantization-based blind watermarks in three different mesh domains: the wavelet domain of a semi-regular mesh, and the spatial and spectral domains of a general mesh. The watermarking primitives, which are subject to scalar Costa quantization, are respectively the norms and orientations of the wavelet coefficient vectors, the analytic volume moments and the manifold harmonics spectral amplitudes. Finally, we detail the design and implementation of a robust mesh watermark benchmarking system, which has been made publicly available on-line. This benchmarking system comprises a standard mesh data set, a software tool and two application-oriented evaluation protocols. The robust mesh watermarking schemes proposed in this thesis and a state-of-the-art method are compared within this benchmarking framework. The comparison results demonstrate both the effectiveness of our blind watermarking schemes and the relevance of our benchmarking system.

Keywords: 3-D mesh, blind watermarking, scalar Costa scheme, wavelet, volume moment, manifold harmonics, benchmark.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Mesh Watermarking and its Applications

Nowadays, three-dimensional (3-D) models are more and more used in applications such as medical imaging, scientific simulation, cultural heritage, digital entertainment and computer-aided design, mainly due to the processing capability improvement of ordinary PCs and the bandwidth increase of network infrastructure. A 3-D model is often numerically represented as a mesh, which is a collection of polygonal facets targeting to constitute an appropriate piecewise linear approximation of the surface of a real 3-D object (c.f. Figure 1.1). A mesh contains two kinds of information: the *geometry* information which represents the 3-D coordinates of its comprised vertices, and the *connectivity* information which describes the adjacency relationship (i.e. edges) between the vertices. Although there exist many other 3-D representations (e.g. implicit surface, NURBS or voxel), 3-D mesh has become the *de facto* standard of numerical representation of 3-D objects due to its algebraic simplicity and high usability. Moreover, it is relatively easy to convert other representations to 3-D mesh, which is considered as a low-level, but effective model.

Unfortunately, like digital images and audio/video clips, 3-D meshes can be easily duplicated and redistributed without any loss of quality by a pirate. This illegal behavior infringes the intellectual property of mesh owners and could also do harm to the whole underlying commercial chains. Actually, the generation of mesh models, either by scanning real 3-D objects or by using specific design software, is normally a time-consuming and expensive work. The robust watermarking technique appears as a good solution to the copyright protection problem of 3-D mesh models. This technique embeds a piece of copyright-related information, i.e. the *watermark*, into the functional part

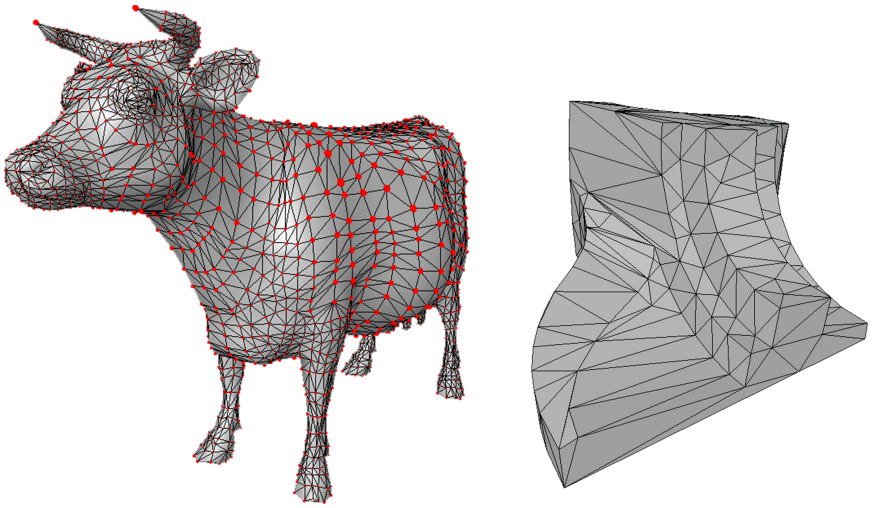


Figure 1.1: Two example meshes: on the left is the Cow model, on the right is the Fandisk model.

of a mesh file. The embedded watermark should be *robust* against various operations and attacks on the watermarked model and also be *imperceptible* to the human visual system. The watermark may be, for instance, the digital identifier of the company who holds the property of the original mesh; later in the case of copyright disputation, the company can extract the embedded watermark to justify its rightful ownership over the model. Besides the robust watermark used for intellectual property protection, *fragile* and *high-capacity* mesh watermarks also have many potential applications such as mesh authentication and content enhancement. A fragile watermark is intentionally designed to be vulnerable to certain attacks. The extraction failure of this watermark indicates the existence of an attack on the watermarked model. A high-capacity watermark is capable of conveying a large number of bits. In general, the purpose of embedding such a watermark is simply to hide a large amount of auxiliary information (e.g. a related website address or an indexing tag) in the original content, so as to enhance its utility or to provide an additional service.

Indeed, since the seminal work of Ohbuchi et al. [OMA97] published in 1997, there has been an increasing interest in the research of 3-D mesh watermarking (c.f. Figure 1.2). We can imagine the following application scenarios of different mesh watermarking schemes.

- An automobile constructor can insert its digital identifier into the car parts it has designed, and this watermark can later be extracted to prove its copyright ownership over the parts.
- A mesh file purchaser can authenticate the integrity and the origin of the received model according to the fragile watermark extraction result. If the watermarking

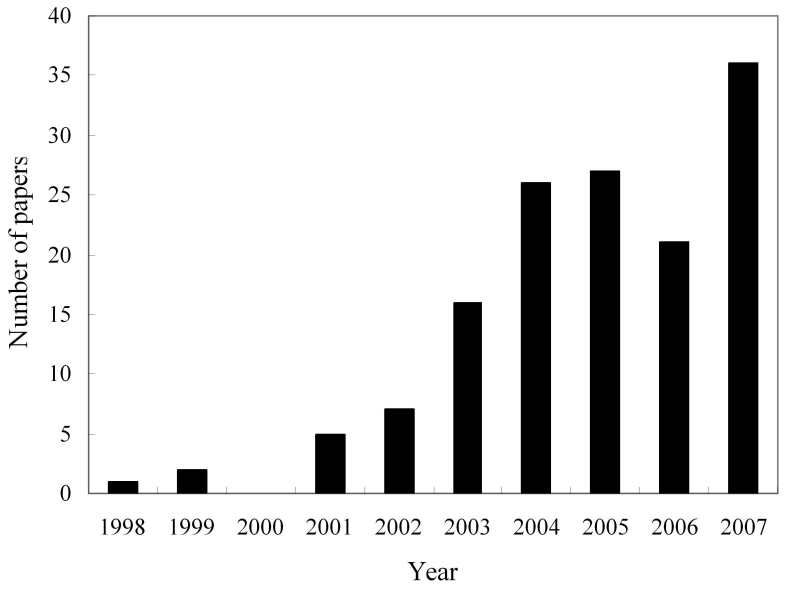


Figure 1.2: The increasing number of research papers on mesh watermarking in EI Compendex.

scheme is well designed, it may also help to locate and fix the tampered parts.

- A doctor can hide a patient's personal information into the 3-D mesh model obtained after a scan through high-capacity watermarking (under condition that this embedding does not impact the diagnosis) to avoid mismatching the patient's personal information and his scan result.
- The texture of a mesh model or even the motion parameters of a mesh sequence could be embedded in the mesh itself via high-capacity watermarking for the purpose of compression, just like hiding a video's audio signal within its visual part.

1.2 Objectives and Contributions

The research topic of this thesis work is the digital watermarking of 3-D polygonal meshes. Our main objective is to construct several effective *blind* mesh watermarking schemes, which do not need the original non-watermarked mesh for the watermark extraction. We focus on the research of blind schemes because they have a much larger practical application range than the non-blind schemes. Indeed, in real-world setting, the original content cannot always be or even should not be present at the watermark extraction phase, often due to algorithm efficiency and security issues. For instance, in the copy control examination application, it is inappropriate to make the original copy available in the control device that is probably in the hand of a malicious client. It is

also meaningless to design a non-blind fragile watermark for content authentication, because the authentication task becomes trivial or even unnecessary if the receiver has the original version. Concerning the work on robust and blind mesh watermarking, we wish to find some approaches to achieving the robustness against *connectivity attacks*, which are deemed to be very destructive to the embedded blind mesh watermarks. Different from *geometry attacks* that only modify the vertex coordinates of the watermarked mesh, connectivity attacks also modify the mesh's connectivity information. Typical examples of such attacks include surface simplification, subdivision and remeshing. In these operations, original vertices, edges and facets may be removed from the watermarked mesh while new vertices, edges and facets may be inserted into it. In practice, it is very difficult to devise a blind mesh watermarking scheme that is capable of resisting connectivity attacks. More details about the different kinds of attacks on watermarked meshes will be presented later in Section 3.3.

In order to accomplish the main objective of this dissertation work, we employ the scalar Costa scheme (SCS) [EBTG03], which is a widely used quantization-based data embedding technique for image, audio and video watermarking, to embed multi-bit blind watermarks in different mesh domains. The exploited mesh domains include the wavelet domain of a semi-regular mesh, and the spatial and spectral domains of a general mesh. The selected watermarking primitives are respectively the wavelet coefficient [LDW97], the analytic volume moment [ZC01] and the manifold harmonics spectral amplitude [VL08]. In the wavelet-based method, three different blind watermarks (robust, high-capacity and fragile) are simultaneously embedded in a same semi-regular mesh without any interference between them. The proposed moment-based and spectral-domain-based blind schemes possess a strong robustness against connectivity attacks and meanwhile a high watermark imperceptibility.

Besides the derivation of some new blind schemes, we would like to bring another contribution to the research community by providing a benchmark for the robust mesh watermarking techniques. The objective of this benchmarking work is to facilitate the evaluation and comparison of different robust methods. The constructed benchmarking system has been made publicly available on-line at `http://liris.cnrs.fr/meshbenchmark/`.

1.3 Outline

The remainder of this manuscript is organized as follows.

Chapter 2 presents some background knowledge on 3-D mesh and digital water-

marking, which is necessary to understand the following chapters.

Chapter 3 provides a comprehensive survey on the research of 3-D mesh watermarking, with an original attack-centric investigation of the state of the art.

Chapter 4 briefly introduces the scalar Costa quantization scheme, which will be used as the basic data embedding technique in our blind mesh watermarking schemes proposed in Chapters 5-7.

Chapter 5 presents a hierarchical multiple watermarking framework for semi-regular meshes which is based on the wavelet transform. Three different blind watermarks are embedded at different appropriate resolution levels of the original semi-regular mesh, through modifications of the corresponding wavelet coefficients of these selected levels.

Chapter 6 describes a robust and blind spatial watermarking technique for general meshes. The strong robustness of the proposed scheme relies on the stability of the mesh's analytic and continuous volume moments.

Chapter 7 presents a blind spectral mesh watermarking scheme that is both computationally efficient and robust against connectivity attacks. The watermarking primitives are the low-frequency mesh spectral amplitudes obtained by means of the recently proposed manifold harmonics analysis.

Chapter 8 describes a benchmark for the robust mesh watermarking methods, which includes a "standard" mesh data set, a software tool and two application-oriented evaluation protocols.

Chapter 9 summarizes the contributions of this dissertation work, draws conclusions of the manuscript and proposes several future working directions concerning the research on 3-D mesh watermarking.

Chapter 2

Background Knowledge on 3-D Mesh and Digital Watermarking

Contents

In this chapter, we first explain some key concepts about the 3-D polygonal meshes. Then, the classification and evaluation metrics of the digital watermarking techniques are presented. Note that here we only provide the minimum necessary background knowledge for the understanding of the remainder of this manuscript, interested readers could refer to additional references ([BPK*07] on 3-D mesh and [KP00, BB04, CMB*07] on digital watermarking) for some further reading.

2.1 Polygonal Mesh

A 3-D polygonal mesh has three different kinds of combinatorial elements: *vertices*, *edges* and *facets* (typically triangles or quadrangles). The coordinates of the vertices constitute the *geometry* information of the mesh, while the edges and facets describe the adjacency relationships between vertices and constitute the mesh's *connectivity* information. Mathematically, a mesh $\mathcal{M}$ containing N_V vertices and N_E edges can be modeled as a signal $\mathcal{M} = \{\mathcal{V}, \mathcal{E}\}$, where

$$\mathcal{V} = \{v_i = (x_i, y_i, z_i) \,|i \in \{1, 2, ..., N_V\}\}\,, \tag{2.1}$$

$$\mathcal{E} = \left\{e_j := \left(p_1^{(j)}, p_2^{(j)}\right) |j \in \{1, 2, ..., N_E\}\,; p_1^{(j)}, p_2^{(j)} \in \{1, 2, ..., N_V\}\right\}. \tag{2.2}$$

More precisely, each vertex v_i is described by its 3-D coordinates (x_i, y_i, z_i); each element in $\mathcal{E}$ stands for an edge connecting two different vertices indexed respectively by $p_1^{(j)}$ and $p_2^{(j)}$. Figure 2.1 shows an example of 3-D mesh. As illustrated by the close-up on the right part of this figure, the *valence* of a vertex is the number of its incident edges and the *degree* of a facet is the number of its component edges. We also define the 1-*ring neighbors* of a vertex as the vertices that are directly connected to it by a certain edge. Formally, for a vertex v_i, its 1-ring neighbors are the elements of the following set:

$$\{v_l | v_l \in \mathcal{V},\ l \neq i,\ \text{and}\ (l, i) \in \mathcal{E}\}\,. \tag{2.3}$$

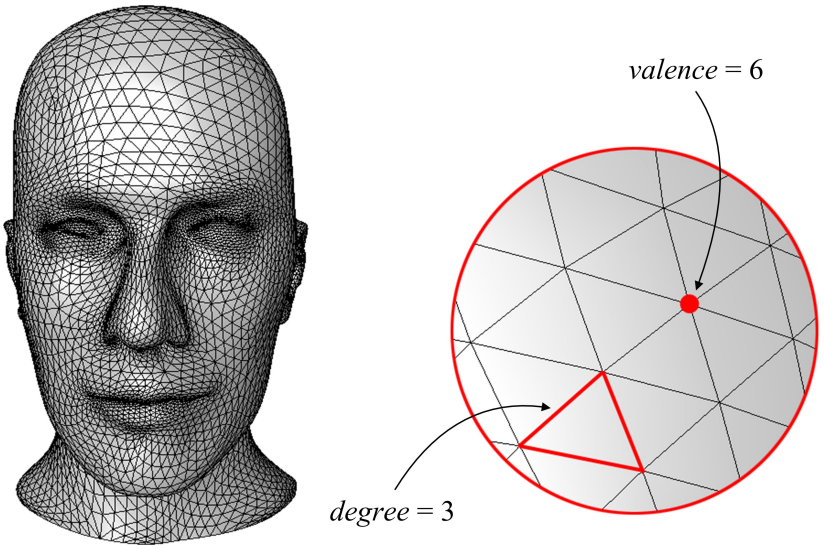
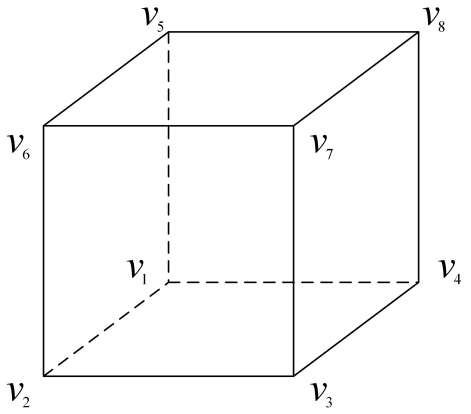


Figure 2.1: The Mannequin mesh model (on left) and a close-up of this model illustrating the concepts of vertex valence and facet degree (on right).

Instead of the list of edges $\mathcal{E}$, the connectivity information of the mesh $\mathcal{M}$ can also




```
v 0.0 0.0 0.0
v 1.0 0.0 0.0
v 1.0 1.0 0.0
v 0.0 1.0 0.0
v 0.0 0.0 1.0
v 1.0 0.0 1.0
v 1.0 1.0 1.0
v 0.0 1.0 1.0

f 1 4 3 2
f 1 2 6 5
f 2 3 7 6
f 3 4 8 7
f 1 5 8 4
f 5 6 7 8
```

Figure 2.2: An example of OBJ mesh file that represents a cube.

be completely described by a list of its N_F facets as

$$\mathcal{F} = \left\{ f_k := \left(p_1^{(k)}, p_2^{(k)}, ..., p_D^{(k)} \right) | k \in \{1, 2, ..., N_F\} \right\}, \tag{2.4}$$

with D the degree of the facet f_k and

$$p_d^{(k)} \in \{1, 2, ..., N_V\}, \left(p_{d-1}^{(k)}, p_d^{(k)} \right) \in \mathcal{E};\ d \in \{1, 2, ..., D\}, p_0^{(k)} := p_D^{(k)}. \tag{2.5}$$

It can be seen that each facet from the list $\mathcal{F}$ is represented by a sequence of indices of its component vertices that are sorted in a certain cyclic order around it. A number of different mesh storage formats have been established, such as the 3-D Object File Format (OFF), the wavefront OBJect format (OBJ), the Stanford University PoLYgon format (PLY) and the Virtual Reality Modeling Language (VRML). All these formats adopt a similar strategy to store a mesh in a raw (uncompressed) way, in the form of a vertex list followed by a facet list. The vertex list is mainly composed of the vertex coordinates. The facet list is mainly composed of the indices of the facet component vertices. Figure 2.2 illustrates an example OBJ file that represents a cube.

A mesh is called *triangular* if all its facets are triangles; similarly we can define a *quadrangular* mesh. A mesh is *regular* if all its vertices have a same valence. A *semi-regular* mesh is a piecewise regular structure and consists of a patchwork of large regular regions; hence, it owns regular vertices almost everywhere. Otherwise, we say that the mesh is *irregular*. A mesh is called *manifold* if the neighborhood of every vertex is homomorphic to a disk or a half-disk. The *orientation* of a facet is defined according to the cyclic order of its component vertices combined with the right hand rule. Obviously, there exist two possibilities for this orientation. The orientations of two adjacent facets are called *compatible* if the two shared vertices are in opposite orders in these two facets.

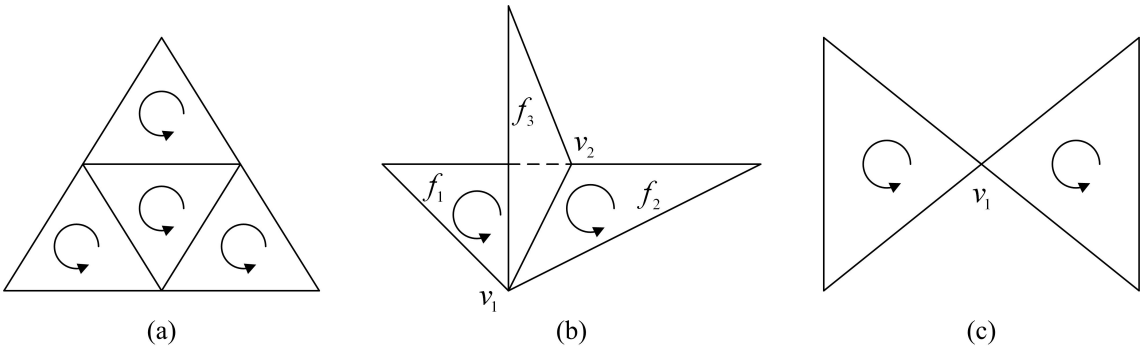


Figure 2.3: Some example meshes to illustrate the concepts of "manifold" and "orientable": (a) a manifold and orientable mesh; (b) a non-manifold and non-orientable mesh; (c) a non-manifold but orientable mesh.

The entire mesh is called *orientable* if we can find a combination of the orientations of all its facets such that each pair of adjacent facets in the mesh is compatible. Figure 2.3.(a) shows an example of manifold and orientable mesh. The mesh illustrated in Figure 2.3.(b) is non-manifold since the neighborhoods of v_1 and v_2 are not homomorphic to a disk or a half-disk; this mesh is also non-orientable because after fixing a pair of compatible orientations for f_1 and f_2, it is impossible to find an orientation for f_3 that makes f_3 compatible with both f_1 and f_2. Figure 2.3.(c) shows a mesh that is orientable but non-manifold (there is a singularity in the neighborhood of v_1).

For a manifold and orientable mesh, the Euler's formula describes the relationship between the numbers of its combinatorial elements and its topological information as follows:

$$N_V - N_E + N_F = 2\,(s - g) - b := \chi, \tag{2.6}$$

where s stands for the number of connected components in the mesh, g is called the *genus* of the mesh, b is the number of mesh borders, and χ is defined as the *Euler's characteristic* of the mesh. In general, the genus of a 3-D object reflects more or less its topological complexity and is intuitively equal to the number of handles in the object. Therefore, a sphere is of genus 0, a torus is of genus 1 and a double torus is of genus 2. Now, suppose that a manifold triangular mesh has sufficient edges and facets and that the number of its border edges is negligible compared to that of its non-border edges, then we have approximately the relationship $2N_E = 3N_F$, since each triangular facet has three edges and most of the mesh edges are shared by two facets. From Equation (2.6) and the above relationship, we can deduce that the formula $2N_V - N_F = 2\chi$ approximately holds. Furthermore, since χ is typically very small for an ordinary mesh, we can then obtain $N_F \approx 2N_V$, which means that the facet number of a typical triangular manifold mesh is twice as its vertex number.

For more background knowledge on polygonal meshes and particularly their applications in geometry processing, readers could refer to the Siggraph course notes of Botsch et al. [BPK*07].

2.2 Digital Watermarking

As mentioned in Section 1.1, the basic idea of digital watermarking techniques [KP00, BB04, CMB*07] is to hide a piece of secret information in the functional part of a multimedia content. In watermarking terminology, the original content in which we embed a watermark is often called the *cover* content, while the obtained watermarked content is also called the *stego* content. Compared to cryptography [MvOV01], the digital watermarking technique is able to protect digital works (assets) after the transmission phase and the legal access, because the watermark exists within the protected multimedia content and thus always travels along with it.

2.2.1 Classification

There are different classifications for watermarking algorithms. First of all, we distinguish between *non-blind* and *blind* watermarking schemes depending on whether or not the original cover content is required for watermark extraction. Usually, one hopes to construct a *robust* watermark, which is able to go through common malicious attacks, for intellectual property protection purposes. However, sometimes the watermark is intentionally designed to be *fragile*, even to very slight modifications, in order to be used in authentication applications. In some cases, the purpose of embedding a watermark is simply to hide a large amount of auxiliary information in the cover content, so as to reinforce its usability. We call this type of watermark as *high-capacity* watermark since in such applications it is often required that the embedded watermark should convey a message of a large number of bits. Researchers also classify watermarking algorithms as *spatial/temporal-domain-based* methods or *transform-domain-based* methods, according to the watermark embedding space. Finally, watermarking schemes can also be classified into *detectable* schemes or *readable* schemes. In a detectable scheme, at the watermark detection side, we can only tell whether the multimedia content under inspection has been watermarked or not. This means that we only have a binary output concerning the existence of a certain watermark within a certain multimedia file. In a readable scheme, the watermark extraction algorithm can decode and output a multi-bit message conveyed by the watermark.

At this point, it seems necessary to explain in more depth the definitions of the different types of watermarking schemes. These key terms will be used afterward throughout the manuscript. In the following explanations, some concepts and notations are extracted from [BB04].

Robust watermarking

Suppose that we want to embed a watermark w in a cover asset A. The watermark w is normally either a bit string or a pseudo-random sequence of a certain statistical distribution. The watermark embedding/insertion process can be modeled as $A_w = \mathcal{I}(w, A, K_I)$, where $\mathcal{I}$ is the embedding function and K_I is a secret key used to ensure the watermarking security (the definition of watermarking security will be explained later in Section 2.2.2). Due to the existence of attacks or processing, the watermarked asset A_w could have been transformed to $\hat{A}_w$ at the extraction/detection side. At this point, we make two assumptions: 1) the distortion between A_w and $\hat{A}_w$ can be measured by using a certain metric and the obtained distortion value is denoted by dis; 2) the maximum tolerable distortion on the watermarked content under a certain application scenario is prescribed as D_{max}. The second assumption means that an attacked asset $\hat{A}_w$ with a distortion greater than D_{max} is considered too degraded to be used in the application. If for any $dis < D_{max}$ caused by attacks or processing, the embedded watermark can be successfully extracted/detected with a sufficient level of confidence, then we say that the watermarking algorithm is robust. Robust watermarking is usually used in intellectual property protection applications. The embedded robust watermark can be extracted and employed, for instance, to demonstrate rightful ownership of the multimedia file, to trace malicious client who illegally redistributes the content or to carry out copy control examination.

Fragile watermarking

Fragile watermarking schemes are mainly used in authentication applications, which aim to verify the integrity of multimedia contents. Here, we assume that the maximum tolerable distortion between A_w and $\hat{A}_w$ under a certain authentication scenario is prescribed as D_{tol}. Indeed, any induced distortion smaller than D_{tol} is considered harmless to the application (i.e. A_w and $\hat{A}_w$ are thought as the same content with respect to the application), while any distortion greater than D_{tol} is considered as non-tolerable (i.e. the difference between A_w and $\hat{A}_w$ is unacceptable with regard to the application). For a fragile watermarking algorithm, it is expected that for any $dis < D_{tol}$ the algorithm can successfully extract the embedded watermark without any error. However, for any $dis \geq D_{tol}$, the algorithm should extract a watermark with some amount of errors, based

on which it can indicate to users the existence of an attack. A well-designed fragile watermarking scheme is also capable of locating or even identifying the endured attack according to the watermark extraction result. Compared to the conventional authentication based on cryptography, the authentication based on fragile watermarking has two main advantages: the capability of distinguishing between tolerable and non-tolerable operations and the possibility of locating and identifying the endured attacks.

High-capacity watermarking

The *capacity* signifies the number of bits conveyed by the inserted watermark w. A high-capacity watermarking scheme allows the embedding of a large number of bits in the cover asset A. In this case, the watermark w is normally a bit string that represents a meaningful message. It is also expected that the embedded watermark possesses a certain level of robustness. This means that when the attack-induced distortion dis is smaller than an application-related threshold D_{max}, we can always successfully extract and decode the embedded bit string w. Compared with the straightforward data attachment in the header of the multimedia file, the advantages of the high-capacity watermarking are that the embedded information is independent of the file format and that it may also be able to survive the digital-to-analog and analog-to-digital conversions.

Detectable and readable watermarking

In a detectable watermarking scheme, we can only verify whether a specific watermark w_s exists in a certain multimedia asset. The verification process normally outputs a Boolean variable as the final result: the output '0' means that w_s is not detected in the asset, while the output '1' signifies the opposite. For this reason, researchers also call detectable watermarking as 1-*bit* watermarking. Note that along with the above Boolean output, the detection algorithm may also provide a floating number which represents the confidence level or correctness probability of the Boolean decision it makes. On the contrary, in a readable watermarking scheme, the embedded watermark w is a multi-bit string. During the watermark retrieval stage, the inserted bit string is extracted and decoded from a stego and potentially attacked asset. It is easy to build a detectable scheme from a readable scheme by simply adding a verification stage after watermark (i.e. a bit string) extraction and decoding. However, it is difficult or even impossible to construct a readable scheme from a detectable scheme. Obviously, readable watermarking schemes have a larger application range than detectable watermarking schemes.

Non-blind, blind and semi-blind watermarking

If we need the original non-watermarked asset A to extract/detect the inserted water-

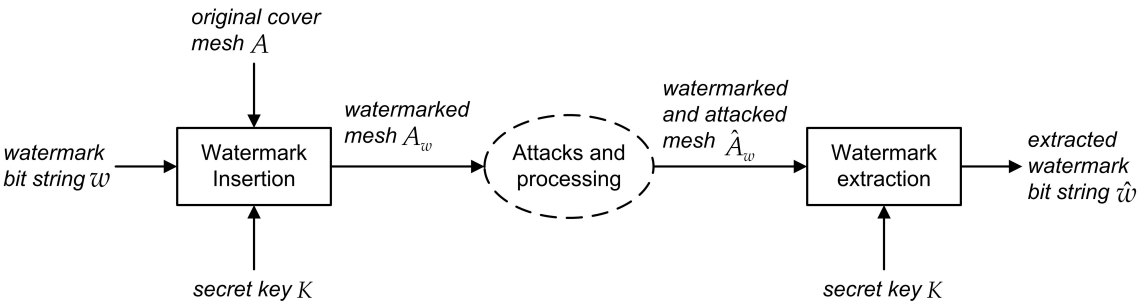


Figure 2.4: The general block diagram of the blind and readable mesh watermarking schemes.

mark, then we say that the algorithm is non-blind. Non-blind watermark extraction (for readable schemes) can be modeled as $\hat{w} = \mathcal{E}(\hat{A}_w, A, K_E)$ and non-blind watermark detection (for detectable schemes) can be modeled as $R = \mathcal{D}(\hat{A}_w, A, w_s, K_E)$. In these two expressions, $\mathcal{E}$ and $\mathcal{D}$ are respectively the watermark extraction and detection function, K_E is a secret key that can either be equal to the key K_I used during the watermark insertion (*symmetric* watermarking) or be different from K_I (*asymmetric* watermarking), w_s is the specific watermark that we search, and the watermark detection result R is a Boolean variable. Contrarily, if the extraction/detection process does not require the original asset A, then we say that the algorithm is blind. Blind watermark extraction can be modeled as $\hat{w} = \mathcal{E}(\hat{A}_w, K_E)$, and blind watermark detection can be modeled as $R = \mathcal{D}(\hat{A}_w, w_s, K_E)$. A trade-off is the so-called *semi-blind* watermarking, which does not require A for watermark extraction/detection, but instead needs some specific information about A (e.g. brightness histogram or invariant moments in the case where A is a 2-D image). The limit between blind and semi-blind schemes seems always ambiguous. In our opinion, if the watermark extraction/detection process requires only a very small amount of asset-dependent information, say $\leq$ 128 bits, then we still consider the algorithm as a blind scheme. This small amount of supplementary information can be easily integrated in the secret key K_E without bringing inconvenience to the users.

In this thesis, we concentrate on the research of blind and readable watermarking techniques for 3-D polygonal meshes. The general block diagram of this kind of watermarking schemes is illustrated in Figure 2.4. Note that the algorithms under investigation are all symmetric watermarking schemes for which the insertion key is the same as the extraction key, i.e. $K_I = K_E = K$. Concerning the aimed applications, we mainly focus on the robust mesh watermarking used for intellectual property protection. Meanwhile, we also propose some fragile and high-capacity schemes, used respectively for content authentication and content enhancement.

2.2.2 Evaluation metrics

A watermarking system is often evaluated by using the following four metrics: capacity, distortion, robustness and security.

Capacity

As mentioned in the definition of the "High-capacity watermarking" in Section 2.2.1, the watermarking capacity is the number of bits of the hidden message conveyed by the watermark. The capacity is often practically measured by bits per primitive. A watermarking *primitive* is the element/quantity subject to modification in order to embed the watermark. It can be, for example, the image pixel brightness, an image block DCT (Discrete Cosine Transform) coefficient or a mesh vertex coordinate.

Distortion

The distortion measures the difference between the original cover content and its watermarked version. This induced distortion can be measured either *objectively* or *perceptually*. The perceptual distortion measurement is normally considered more important than the objective distortion measurement. The main reason is that in most applications, it is demanded that the user (a human being) should not perceive annoying distortion in the watermarked multimedia content, i.e. the watermark should be *imperceptible* or *unnoticeable* to the human visual system. But unfortunately, the objective distortion measurement, in the case of both images and meshes, does not always correctly reflect the visual difference between two images or meshes. Therefore, in practice, the imperceptibility of the watermark is measured either by subjective evaluation results given by human beings or by a well-designed "objective" perceptual distortion metric. It is worthwhile pointing out that in some mesh applications, such as computer-aided design and medical imaging, the induced objective distortion is more important than the perceptual distortion, because in these applications, a high-level geometric precision is required. Finally, note that in the following, we will also use the term "distortion" to indicate the modification induced by an attack on the watermarked content. These two kinds of distortion can actually be measured by using the same metrics.

Robustness

The robustness indicates how resistant the watermarking scheme is against various routine attacks and processing on the watermarked content. For readable watermarking schemes, the robustness is often measured by some plots or tables that illustrate the relationship between a watermark extraction correctness metric and the amplitudes of the endured attacks. Some commonly used extraction correctness metrics include the bit

error rate, the bit detection ratio and the correlation between the extracted watermark bit string and the initially inserted bit string.

Security

A secure watermarking scheme should be able to withstand the malicious attacks that aim to break down the whole watermarking-based copyright protection or authentication system through, for instance, secret key disclosure or inversion of the watermark embedding procedure. The security is rather considered as a high-level requirement. However, we should at least ensure that it is impossible or computationally very expensive to carry out an unauthorized watermark extraction or an optimal watermark removal. A secret key is required in a watermarking system in order to achieve a minimum level of security. Usually, this key serves to generate the pseudo-random values of some secret parameters of the system which are only known to authorized users.

Note that the requirements on these four metrics are often contradictory. For example, a higher watermark embedding intensity normally leads to a better robustness, but meanwhile may degrade the visual quality of the watermarked content and make the watermark perceptible. Redundant watermark insertion can sometimes considerably strengthen the robustness, but at the same time inevitably decreases the capacity. When building a watermarking system, we should find a satisfactory trade-off between these four important metrics according to the requirements of the aimed application.

After presenting in this chapter some background knowledge on 3-D mesh and digital watermarking, in the next chapter, we will provide a comprehensive survey on 3-D mesh watermarking.

Chapter 3

Survey on 3-D Mesh Watermarking

Contents

This chapter provides a comprehensive survey on 3-D mesh watermarking research. First, the particular difficulties encountered while devising mesh watermarking

methods are explained. Then, we review the existing techniques by classifying them as fragile schemes, high-capacity schemes and robust schemes. An attack-centric investigation of this state of the art is also provided. The attacks on watermarked meshes are classified and analyzed; the existing solutions to resisting each kind of attacks are presented.

Most contents of this chapter were published in an international conference paper [WLDB07b] and in an international journal paper [WLDB08a]. Readers could also refer to the paper of Rondao-Alface and Macq [RAM07] for another survey on 3-D mesh watermarking.

3.1 Difficulties and Classification

So far, there still exist few watermarking methods for 3-D meshes, in contrast with the relative maturity of the theory and practices of image, audio and video watermarking. This situation is mainly due to two difficulties: 1) the intrinsic irregular sampling nature of 3-D meshes, and 2) the complexity of the possible attacks on watermarked models. These two problems will be explained in the following paragraphs.

In the case of 2-D image watermarking, the cover image can be considered as a matrix, and each pixel as an element of this matrix. This means that all pixels have an intrinsic order in the image, for example the order established by row or column scanning. This order is often used to *synchronize* the watermark (i.e. to know where the elements of the watermark signal w are embedded, and in which order). On the contrary, there is no simple, robust and intrinsic ordering for the combinatorial elements of a 3-D mesh, which often constitute the carriers (primitives) of the watermark signal elements. Some intuitive orders, such as the order of the vertices and facets in the mesh file, or the order of vertices obtained by ranking their projections on an axis of the objective coordinate system, are easy to be altered. Moreover, because of their irregular sampling nature, we still lack an effective spectral analysis tool for 3-D meshes. This situation makes it difficult to devise successful spectral mesh watermarking schemes.

In addition to the above point, robust watermarks also have to face various intractable attacks. The reordering of vertices and facets in the mesh file does not have any impact on the mesh shape, but it can desynchronize the watermarks that rely on this straightforward ordering. The similarity transformations, including translation, rotation, uniform scaling and their combination, are supposed to be common operations through which a robust watermark, or even a fragile or a high-capacity watermark, should be able to survive. Moreover, original watermarking primitives (e.g. vertices, edges or facets) could disappear after a mesh simplification or remeshing. These attacks can be easily conducted by using some freely available software tools such as ReMESH [AF06] and MeshLab [CCR08], and they can completely destroy the geometry and the connectivity information of a watermarked mesh while well preserving its global shape. As mentioned in Section 1.2, we distinguish between the *geometry attacks*, which only modify the positions of the vertices, and the *connectivity attacks*, which also change the connectivity aspect of the mesh. Typical geometry attacks include noise addition, smoothing and vertex coordinate quantization. Examples of connectivity attacks are surface simplification, subdivision, remeshing and cropping. Section 3.3 provides a detailed investigation on the attacks on watermarked meshes, and also discusses the

existing solutions in order to make the watermark robust against them.

As presented in Section 1.2, the main objective of this thesis work is to construct some blind and robust mesh watermarking schemes. However, due to the difficulties presented above, it seems quite difficult to accomplish this objective. Indeed, at the theoretical level, it has been proven that the requirement of blindness does not cause any performance loss to a watermarking method, at least under certain assumptions [Cos83]. However, practically, a blind mesh watermark is normally much less robust than a non-blind one. In the non-blind case, the availability of the original cover mesh makes watermark extraction/detection much easier, mainly in sense that it can facilitate the watermark synchronization process, especially under connectivity attacks. Hence, the main difficulties encountered while devising a blind and robust mesh watermark consist in finding an appropriate watermarking primitive and establishing a robust synchronization mechanism.

In the following section, we will present the existing mesh watermarking algorithms by classifying them as fragile, high-capacity and robust techniques. In each class, it seems convenient to subdivide the members into two subclasses, depending on whether the watermark is embedded in the spatial domain or in a transform domain of the cover mesh. In a spatial-domain-based scheme, the watermark is embedded by directly modifying the mesh geometry or connectivity, while in a transform-domain-based scheme, the watermark is inserted through modulation of the spectral-like coefficients obtained after a certain mesh transformation.

3.2 3-D Mesh Watermarking Techniques

3.2.1 Fragile techniques

A fragile watermarking technique used for authentication applications often has to possess two properties: it should be vulnerable to even very slight modifications on the watermarked asset; and it should be capable of locating, or even identifying the endured attacks. However, we often want a (semi-)fragile mesh watermark to be robust against the so-called *content-preserving operations* including vertex/facet reordering in the mesh file (also called *element reordering*) and similarity transformations. In many applications, these operations are not considered as malicious attacks, but as routine operations since theoretically they do not have any influence on the mesh shape. Meanwhile, a fragile mesh watermark should be blind, because in general it is meaningless to construct a non-blind watermarking-based authentication algorithm (c.f. Section 1.2).

3.2.1.1 Fragile techniques in spatial domain

As stated in Section 2.1, the spatial description of a 3-D mesh includes a geometry aspect and a connectivity aspect. We begin with the techniques modifying the geometry.

Fragile techniques in spatial domain modifying the geometry

Yeo and Yeung [YY99] proposed the first fragile mesh watermarking scheme in the literature. Their basic idea is to search for a new position for each vertex where two predefined hash functions have an identical value, in order to make all the mesh vertices valid for authentication. At extraction, one simply examines the validity of each vertex by verifying the equality between the two hash function values, and locates the attacks on invalid vertices. The watermark embedding algorithm depends on the vertex orders that are pre-established in the mesh file, so as to prevent the *causality problem*. Formally, the causality problem means that the embedding of the posterior watermark signal elements (either bits or pseudo-random values of a certain statistical distribution) impacts the synchronization of the anteriorly embedded ones, or directly changes the feature values of the watermarking primitives in which the anterior elements are embedded; hence, the extracted watermark can be different from the original one, even under no attacks. Actually, the causality problem appears quite frequently in 3-D mesh watermarking, especially in fragile schemes and robust schemes. In the algorithm of Yeo and Yeung, the first hash function is dependent only on the position of the current vertex that is to be watermarked, but the second one also depends on the positions of its 1-ring neighbors (c.f. Section 2.1 for the definition of 1-ring neighbors). When considering these neighbors for the calculation of the second hash function value, the authors only take into account the already watermarked ones, which are in front of the current vertex in the pre-established order (Figure 3.1 illustrates a simple example). Without this precondition, the causality problem occurs, which in this case means that the watermark embedding in a certain vertex will impact the validities of its 1-ring neighbors that have already been watermarked at that moment. Hence, the scheme of Yeo and Yeung cannot resist reordering of the vertex indices in the mesh description file.

Lin et al. [LLLL05] considered vertex reordering as an operation that a fragile watermark should be able to resist because it is harmless to the mesh shape. Therefore, they solved the causality problem and at the same time achieved the invariance to vertex reordering by setting both hash functions dependent only on the coordinates of the to-be-watermarked vertex. Chou and Tseng [CT06] solved the causality problem by introducing the concept of *adjusting vertex*. In their algorithm, one of the two hash func-

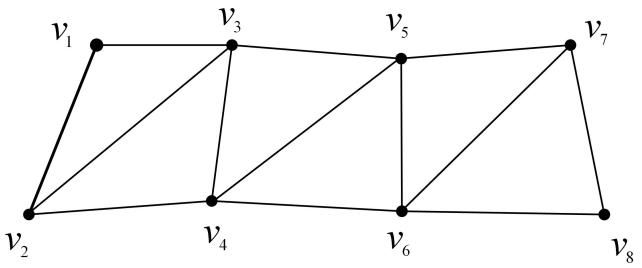


Figure 3.1: This figure illustrates the causality problem encountered and solved in the method of Yeo and Yeung [YY99]. For example, for the calculation of the second hash function value of the vertex v_3, we only take into account the vertices v_1 and v_2, which have already been watermarked at that moment. If v_4 and v_5 were also involved, then after the watermarking of v_4 and v_5 through modification of their coordinates, the validity of v_3 would probably be altered because its second hash function value would be changed.

tions is dependent on the barycenter of the vertex's 1-ring neighbors. However, nearly every watermarked vertex has an adjusting vertex selected from its neighbors. The positions of the adjusting vertices are tuned after the displacement of the watermarked vertices, in order to recover the barycenter of the neighbors of each watermarked vertex to its original value. Another feature is that the displacement upper-bound for watermarked vertices in their scheme is accurately controlled so that severe distortions are avoided. Wu and Chueng [WC06] proposed a fragile scheme by choosing the distance from a vertex to the centroid of its already traversed and watermarked neighbors as the authentication primitive. Their watermark is invariant to similarity transformation but vulnerable to vertex/facet reordering because their vertex traversal algorithm is based on the original vertex/facet indices in the mesh file. Recently, Wang et al. [WZYG08] pointed out that we can avoid the causality problem by simply making the watermarked vertices not adjacent to each other. Based on this idea, they proposed a fragile scheme similar to that of Chou and Tseng. Meanwhile, compared to the other schemes that use floating-point arithmetic [YY99, LLLL05, CT06, WC06], the algorithm of Wang et al. is numerically stable since it only uses bit operation and integral arithmetic to carry out watermark embedding and extraction. However, like the other methods that are based on hash functions [YY99, LLLL05, CT06], this scheme is not immune to similarity transformation.

Fragile techniques in spatial domain modifying the connectivity

Ohbuchi et al. [OMA97] presented two visible mesh watermarking algorithms based on connectivity modification. In the first algorithm, the local triangulation density is

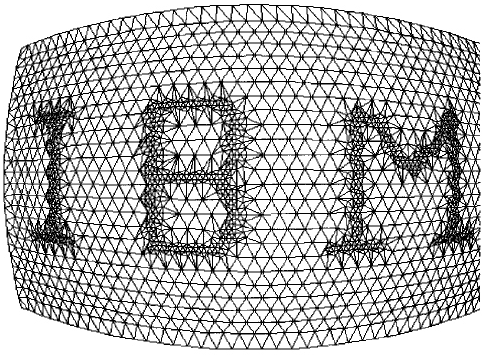


Figure 3.2: Based on local triangulation density modification, Ohbuchi et al. [OMA97] proposed a visible 3-D mesh watermark. In fact, the geometry of the mesh has also been modified because some vertices were inserted so as to modify the triangulation density. This figure is extracted from [OMA97].

changed to embed a watermark (c.f. Figure 3.2). The second algorithm first cuts one band of triangular facets off the mesh and then glues it to the cropped mesh with just one edge. In these two methods, the embedded watermarks do not spread all over the mesh; this fact, along with their visibility to human eyes, prevents them from being useful fragile watermarks due to the lack of attack localization capability and security.

3.2.1.2 Fragile techniques in transform domain

Usually, researchers choose to operate in a kind of spectral domain with the objective to improve the resistance and/or the imperceptibility of a robust watermark (we will give some explanations on this point at the beginning of Section 3.2.3.2). However, in the case of 3-D meshes, multiresolution analysis seems more flexible than the other spectral-like transforms, in sense that it is possible to construct all types of watermarks (robust, fragile and high-capacity) in the obtained multiresolution domain.

3-D mesh multiresolution analysis [DFS05] is a useful tool to reach an acceptable trade-off between the mesh complexity and the processing, storage or visualization capacity of the available resource. Such an analysis produces a coarse mesh that represents the basic shape (low frequencies) of the model and a set of details information at different resolution levels (medium and high frequencies). During the dual synthesis process, we can obtain a series of reconstructed meshes, all representing a same 3-D object but with different complexities, i.e. resolutions. As mentioned above, the most interesting point of the multiresolution analysis for watermarking is its flexibility: there are different embedding locations that can satisfy different application requirements. For example, embedding in the coarsest representation ensures a good robustness, while embedding in the details parts provides a very good capacity. Under the same additive embedding

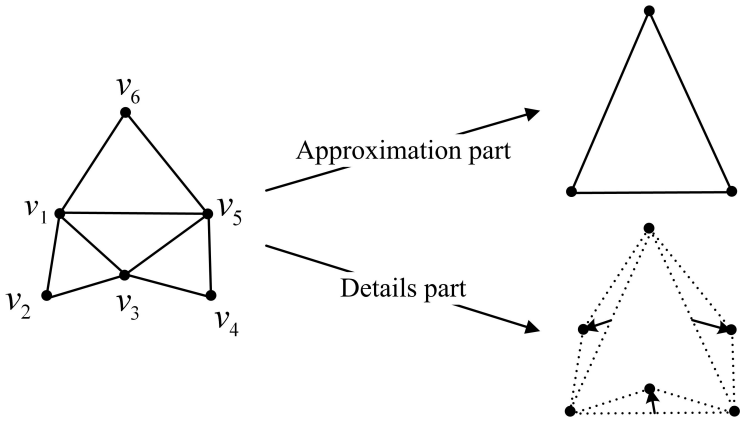


Figure 3.3: Illustration of the lazy wavelet decomposition mechanism for semi-regular triangular meshes.

intensity, embedding in the mesh low resolution component can be both more robust and more imperceptible because it makes the object globally expand or contract a little, while not introducing annoying distortions. Embedding in high resolution levels may lead to the construction of some effective fragile watermarking schemes that are capable of precisely locating the endured attacks.

Wavelets are a common tool for carrying out mesh multiresolution analysis. The mathematical formulation of the wavelet analysis and synthesis of 3-D meshes was introduced by Lounsbery et al. [LDW97]. Figure 3.3 illustrates the principle of the lazy wavelet decomposition mechanism for semi-regular triangular meshes. In each iteration step of the decomposition, a group of four triangles is merged into one triangle and three of the six initial vertices (*even* vertices, v_2,v_4,v_6 in Figure 3.3) are conserved in the lower resolution. The wavelet coefficients are calculated as the prediction errors for all the removed vertices (*odd* vertices, v_1,v_3,v_5 in Figure 3.3) and they are 3-D vectors associated with each edge of the coarser mesh. A straightforward prediction is used here, which is the midpoint of the two even vertices having been incident to the removed odd vertex. Such an analysis can be iteratively applied on a dense mesh with semi-regular connectivity, and the dual synthesis algorithm can accomplish the inverse reconstruction. Figure 3.4 illustrates the wavelet decomposition of a dense Rabbit mesh.

Cho et al. [CLLP05] proposed a fragile watermarking scheme for semi-regular meshes based on the wavelet transform. They first apply several wavelet decompositions on the original dense mesh and then consider the facets in the obtained coarse mesh as authentication primitives. The basic idea, which is quite similar to that of the spatial method of Yeo and Yeung [YY99], is to slightly modify the shape of each facet so that the values of two predefined hash functions on this facet are the same. The

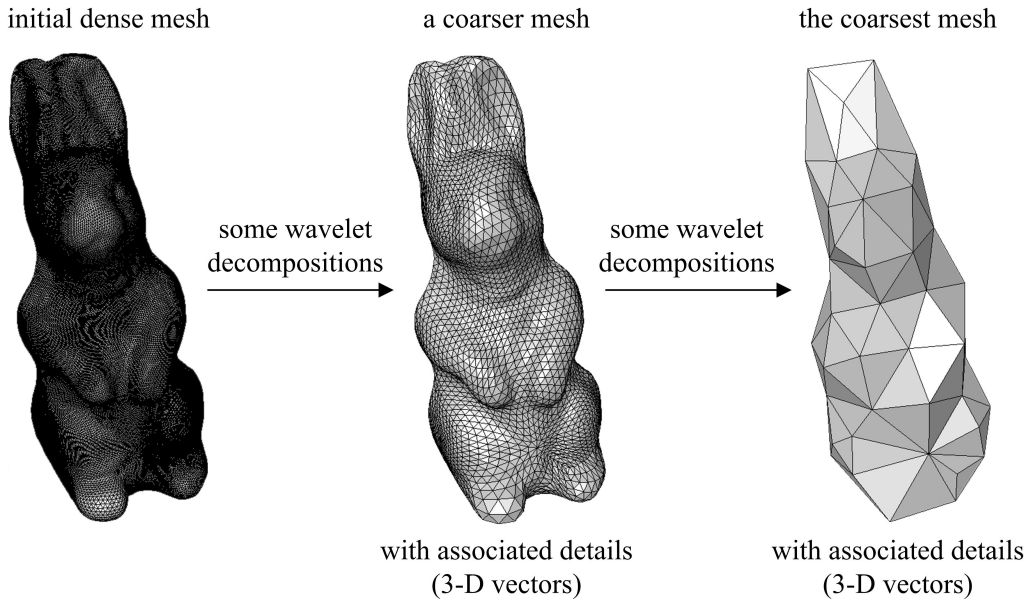


Figure 3.4: Wavelet decomposition of the semi-regular Rabbit mesh.

Table 3.1: Comparison between different fragile mesh watermarking techniques.

Method	Invariance to element reordering	Invariance to similarity transformations	Attack localization
Yeo & Yeung [YY99]	No	No	Yes
Lin et al. [LLLL05]	Yes	No	Yes
Chou & Tseng [CT06]	Yes	No	Yes
Wu & Chueng [WC06]	No	Yes	Yes
Wang et al. [WZYG08]	Yes	No	Yes
Cho et al. [CLLP05]	Yes	Yes	Not precisely

inputs of both functions are invariant to similarity transformations. However, it seems that there exist two problems: first, the causality problem occurs because the modification of a certain facet can influence the validities of its neighboring facets that have already been watermarked, but unfortunately this problem was not discussed by the authors; second, the watermark is embedded in a relatively coarse mesh obtained after several wavelet decompositions, which seems disadvantageous to provide precise attack localization capability. Later in Chapter 5, we will present a new fragile scheme for semi-regular meshes, which does not suffer from the causality problem and is also capable of precisely locating the endured attacks.

Table 3.1 compares the different fragile mesh watermarking techniques. We can see that not any of them is invariant to both element reordering and similarity transformation, while being able to precisely locate the endured attacks.

3.2.2 High-capacity techniques

The objective of a high-capacity mesh watermarking technique is simply to hide a large amount of auxiliary information within the cover model. Sometimes, it is desired that the embedded watermark is invariant to certain kinds of operations, such as element reordering and similarity transformation. Usually, we also require that a high-capacity mesh watermarking scheme should be blind, so as to facilitate the watermark extraction and thus enlarge the application range of the algorithm.

Most high-capacity mesh watermarking techniques are spatial methods and take the individual vertex coordinates as the watermarking primitives. Cayre and Macq [CM03] proposed a high-capacity blind data hiding algorithm for triangular meshes. The chosen watermarking primitive is the projection of a vertex on its opposite edge in a triangle (c.f. Figure 3.5). The capacity of their scheme can attain 1 bit/vertex. The synchronization mechanism relies on the choice of the first triangle according to a certain geometric criterion (e.g. one of the triangles intersecting with the most significant principal axis of the mesh) and a further geometric spreading scheme that is guided by a secret key. A higher capacity, which is about 3 bits/vertex, is achieved by Wang and Cheng [WC05, CW06] by applying a multi-level embedding procedure. This procedure consists of modifying successively the parallel, vertical and rotary positions of a vertex related to its opposite edge in a triangular facet. Cheng and Wang further improved the capacity of their scheme in [CW07] by adaptively embedding more bits in rough regions of the mesh surface where the induced distortions are less visible to human eyes. In the method of Tsai et al. [TWC*06], the neighborhood of a vertex is divided into eight subspaces based on a binary space partitioning tree, and the current vertex is moved into the correct subspace according to the next 3 bits of the watermark sequence. Therefore, the capacity of this method is about 3 bits/vertex.

Some other methods, such as those of Cheng and Wang [CW06] and Bogomjakov et al. [BGI08], modify the orders of the vertices and facets in the mesh file to embed high-capacity watermarks. Particularly, Bogomjakov et al. proposed a fast and sub-optimal variant of the standard permutation steganography [Art01], which can achieve a capacity of $((\log_2 n!) - n + 1)$ bits on a dataset of n elements. For polygonal meshes, such a dataset used for watermark embedding can be the vertex or the facet list. The basic idea of the permutation steganography is that there are $n!$ possible permutations for the n elements in the dataset; therefore, we can hide in maximum a message of $(\log_2 n!)$ bits by properly setting this permutation. These order-based methods have much higher capacities than the geometry-based methods presented in the last paragraph. In addition,

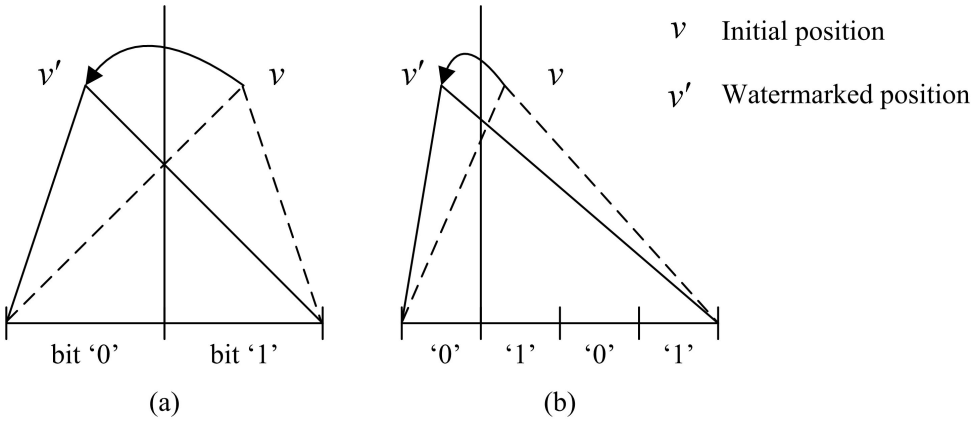


Figure 3.5: Watermarking primitive used in the algorithm of Cayre and Macq [CM03], the projection of a vertex is moved into the nearest correct interval with regard to the watermark bit to be inserted: (a) the opposite edge is divided into two intervals; (b) the opposite edge is divided into four intervals. The inserted bits are both '0'.

they do not introduce any distortion to the cover mesh and also have the advantage of being invariant to geometry attacks. Their main drawback is that the embedded message can also be easily erased by a simple vertex/facet reordering, also without introducing any distortion. Hence, in a practical application, we should carefully choose between geometry-based techniques and order-based techniques, according to the application's robustness requirement, if there is any. In Chapter 5, we propose a wavelet-based high-capacity method in which the watermark is embedded by applying the basic idea of the permutation steganography on a geometric primitive of the cover mesh. Therefore, the proposed method has the good properties of both the geometry-based and the order-based techniques.

Table 3.2 compares the different high-capacity mesh watermarking techniques presented in this subsection. All the techniques compared in this table are blind schemes. Finally, it is worthwhile pointing out that high-capacity watermarks are often fragile (in sense that they are not robust), and some of them have the potential to be successful fragile watermarks with a precise attack localization capability and the invariance to all the content-preserving operations.

3.2.3 Robust techniques

A robust technique should at least be able to resist the attacks that cause distortions smaller than a certain threshold beyond which the attacked stego mesh is greatly degraded. Meanwhile, we also have to ensure that the watermark-induced distortion is objectively and/or perceptually within the tolerance of the aimed application. Concern-

Table 3.2: Comparison between different high-capacity mesh watermarking techniques.

Category	Algorithm	Capacity	Invariance to element reordering	Invariance to similarity transformation
Geometry-based techniques	Cayre & Macq [CM03]	≈ 1 bit/vertex	Yes	Yes
	Wang & Cheng [WC05]	≈ 3 bits/vertex	Yes	Yes
	Cheng & Wang [CW07]	$3 \sim 6$ bits/vertex	Yes	Yes
	Tsai et al. [TWC*06]	3 bits/vertex	Yes	Yes
Order-based techniques	Cheng and Wang [CW06]	≈ 6 bits/vertex	No	Yes
	Bogomjakov et al. [BGI08]	$((\log_2 n!) - n + 1)^*$	No	Yes

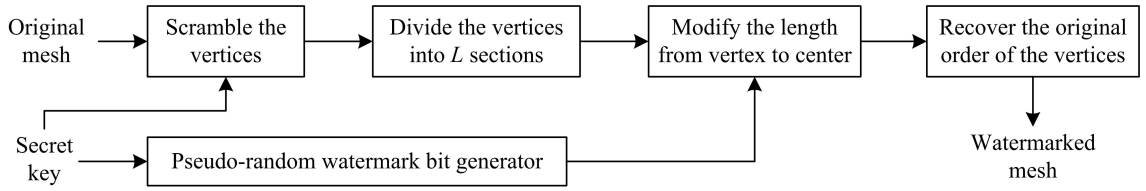
*For a data set of n elements. Practically, the data set can be the mesh's N_V vertices or its N_F facets.

ing the watermarking capacity, for detectable schemes used in the copyright ownership verification application, only 1 bit of information has to be embedded (normally in form of a pseudo-random number sequence). For readable schemes, the capacity depends heavily on the aimed application: for instance, in the copy control application, a low capacity of just a few bits is already sufficient; contrarily, in the typical intellectual property protection application, we usually have to ensure a capacity of about 70 bits, so as to embed the digital identifiers of the content owner, the content purchaser and the asset itself [KP99]. The presentation of the existing robust techniques in this subsection focuses more on the watermarking primitives than on the robustness against various attacks. The latter will be discussed in detail in Section 3.3.

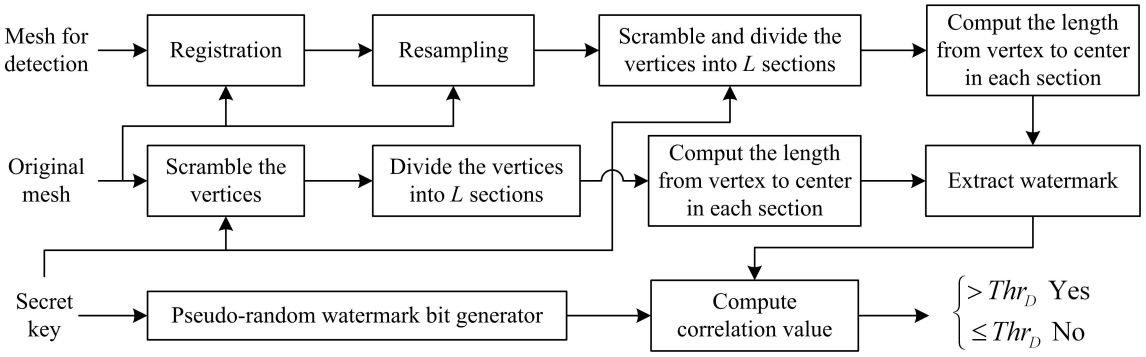
3.2.3.1 Robust techniques in spatial domain

Between the geometry and the connectivity parts of a 3-D mesh, nearly all the existing spatial robust algorithms take the former as the watermarking primitive. Indeed, the fragility to connectivity attacks of the algorithms modifying the mesh's connectivity information prevents them from being effective (blind) robust schemes. Note that some techniques presented in this subsection are not strictly robust, but they are neither fragile. These techniques can be considered as data hiding schemes that were proposed during early stage of the development of 3-D mesh watermarking techniques.

As observed in Section 3.2.1.2 concerning the fragile techniques in the spatial domain, inserting 1 bit in each vertex makes the embedded watermark very vulnerable. Therefore, some algorithms choose the positions of groups of vertices as watermarking primitives with the objective to improve the robustness. Yu et al. [YIK03] presented a non-blind robust algorithm. Figure 3.6 illustrates the main steps of their algorithm. Mesh vertices are first scrambled and divided into several groups using a selected secret key, and then in each of these groups one bit is embedded by modifying the lengths from its member vertices to the mesh center. The modulation mechanism is a simple



(a) Mains steps of watermark insertion



(b) Mains steps of watermark verification

Figure 3.6: Illustration of the main steps of the watermark insertion and verification procedures in the algorithm of Yu et al. [YIK03]. The watermark is first extracted and then verified. For this purpose, we first calculate a correlation value between the extracted watermark bit string and the bit string to verify. Then, we compare this correlation value with a threshold Thr_D to make the verification decision. This method is also an example of constructing a detectable watermarking scheme from a readable watermarking scheme (strictly speaking, it is readable if the watermark signal is a meaningful message instead of a pseudo-random bit sequence).

additive scheme with an adaptive intensity derived from a local geometric analysis of the cover mesh surface. The watermark extraction is also quite simple, since it just needs to regroup the vertices and to inverse the additive watermark bit insertion model. However, a step of registration and resampling is necessary before the watermark extraction, in order to ensure a satisfactory robustness and to recover the same grouping of the vertices at extraction as during the insertion. This pre-processing step needs the original cover mesh and thus makes the algorithm non-blind. The good robustness of this method relies on the non-blind extraction and on the redundant embedding of the same watermark bit in every member vertex within a group.

Similarly, in the "Vertex Flood Algorithm (VFA)" proposed by Benedens [Ben99b], after grouping the mesh vertices according to their distances to the center of a designated triangle, each group's range interval of this "vertex-to-triangle" distance is then divided into $m = 2^n$ subintervals. The distances between all the member vertices in a same group and the chosen triangle center are then modified so that the new distances all fall into a same subinterval that correctly stands for the next n watermark bits.

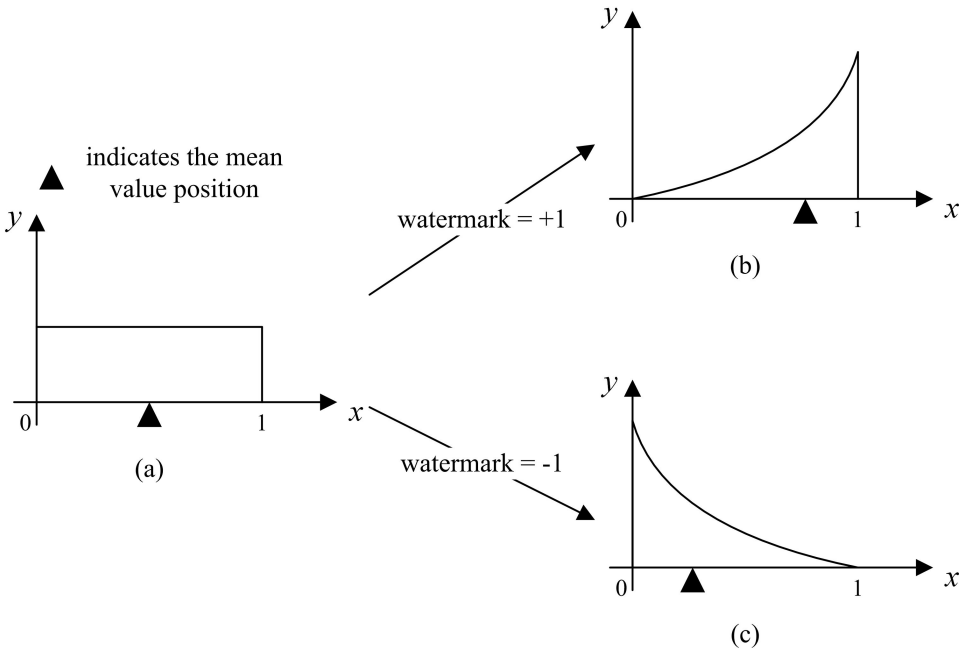


Figure 3.7: Watermark embedding in the algorithm of Cho et al. [CPJ07] that modifies the mean value of the histogram in a bin: (a) the assumed uniform distribution in a bin; (b) the mean value is increased to embed a bit '+1'; (c) the mean value is decreased to embed a bit '−1'. The horizontal axis indicates the normalized distances from vertices to the mesh center (normalized relative vertex norms), and the vertical axis represents the occurrence probability.

Recently, researchers have attempted to embed watermarks in certain kinds of shape histograms. Zafeiriou et al. [ZTP05] first calculate the center and principal axes of the mesh object and afterwards convert the vertex coordinates into the registered spherical coordinate system (r, θ, φ), then they divide the vertices into several groups associated with different ranges of θ. The histogram of the prediction errors of the vertex radial components is constructed for each group. The prediction is calculated from the vertex 1-ring neighbors by applying a local neighborhood operator. The authors assume that the obtained histograms are of Gaussian distribution, and then embed one bit in each vertex group by modifying the shape of this distribution. The idea is to change the one-side variance of the assumed Gaussian distribution either on the left or on the right so as to indicate the bit '−1' or the bit '+1', respectively. Similarly, Cho et al. [CPJ07] construct the histogram of the distances between vertices and the mesh center, and then divide this histogram in bins associated with different ranges of this distance. They make the hypothesis of a uniform histogram distribution in the obtained bins. One bit is embedded in each bin by slightly modifying either the mean value (c.f. Figure 3.7) or the variance of the distribution in the bin.

Both histogram-based methods [ZTP05, CPJ07] are blind and robust against most

kinds of attacks except for cropping and anisotropic connectivity changes, mainly because the calculation of the mesh center and principal axes in these two methods is not stable under such attacks. Moreover, as in the case of image watermarking, the methods based on histogram modification normally have a low security level. Also note that the two schemes have different strategies to achieve invariance to similarity transformations, either by carrying out a blind and robust mesh registration at extraction to recover the same canonical pose as that used during watermark embedding [ZTP05], or by using an invariant watermarking primitive, i.e. the distribution of the distances from vertices to mesh center [CPJ07]. However, these methods may suffer from the causality problem because the key parameters during the histogram reconstruction, such as the mesh center in both methods and the principal axes in the method of Zafeiriou et al., could have been changed after watermark embedding. Unfortunately, neither of these two papers has discussed the impact of this problem on the algorithm's robustness. Nevertheless, the basic idea of their algorithms deserves deeper investigation because the statistical mesh shape features represented in these histograms are quite robust and can be very good watermark carriers.

Furthermore, watermark embedding in the spherical coordinate system, especially in the radial component $r_i = \sqrt{x_i^2 + y_i^2 + z_i^2}$, has additional advantages. We can easily devise some similarity-transformation-invariant algorithms if the distance component is relative to the mesh center, such as in the method of Cho et al. [CPJ07]. Moreover, since the component r_i represents approximately the mesh shape, its modification is supposed to be more robust than a single x_i, y_i or z_i component modification. Hence, numerous researchers have chosen to embed watermarks in the spherical coordinate system [AEJ04, ME04, LS05, ZTP05, CPJ07, LK08].

Facets have several interesting measures for watermarking. In the Triangle Similarity Quadruple (TSQ) algorithm proposed by Ohbuchi et al. [OMA98], the watermarking primitive is the ratio between the height of a triangle and its opposite edge length. TSQ algorithm is blind and intrinsically invariant to similarity transformation. Benedens [Ben99b] described a blind data hiding algorithm in which the triangular facet height is quantized.

Besides the facet shape, the normal direction of the facet has also been used for mesh watermarking. Benedens [Ben99a] proposed a robust method based on the mesh's Extended Gaussian Image (EGI) [Hor84]. First, the EGI of the cover mesh is established by mapping the normal vectors of the mesh facets onto a unit sphere (called the Gaussian sphere) and by assigning to each point on the Gaussian sphere a weight that is equal to

the total area of the facets having the given normal. Then the mesh facets are clustered into several bins according to their normal directions, through surface discretization of the Gaussian sphere on which the mesh EGI has been established. After that, in each bin, the average normal direction of its component facets is modified to hide one bit. Since these average normal directions roughly describe the shape of the mesh, this scheme is proven to be relatively robust against surface simplification and remeshing. Later, Kwon et al. [KKL*03] presented a similar watermarking approach based on EGI. The watermark is embedded through modification of the distribution of the normal vectors of the facets in a same EGI bin. Instead of EGI, Lee and Kwon [LK07] adopted Complex EGI for mesh watermarking. They carry out the mapping of the facet normal direction and the clustering of the facets in the same way as Benedens and Kwon et al., but associate each bin with a complex weight, which depends not only on the total surface of the bin's component facets but also on the proximity of these facets. In their algorithm, the bins having complex weights of bigger amplitudes are selected as watermark bit carriers, and this selection is proven effective to reinforce the robustness. All the above three algorithms [Ben99a, KKL*03, LK07] need to recover the original mesh pose in 3-D space at the extraction phase, in order to achieve an invariant EGI or Complex EGI. For this purpose, some feature values about the cover mesh have to be transmitted to the watermark extraction side. This constraint makes the three algorithms semi-blind.

One inconvenience of the facet-based algorithms is that the modification of the positions of the involved vertices is indirect and sometimes quite complicated, especially in the last three algorithms based on EGI or complex EGI [Ben99a, KKL*03, LK07]. In general, one important motivation of embedding watermark in the mesh facets is to reinforce the robustness, especially against similarity transformation or simplification. However, the final modification on vertices is indirect, and it is sometimes difficult to control the introduced distortion and the expected robustness.

There exist other spatial watermarking techniques that modify the mesh geometry, which are not quite robust but all have some particularity worth mentioning. Harte and Bors [HB02, Bor06] presented a blind algorithm that is robust against similarity transformation. The primitive is the relative position of a vertex to its 1-ring neighbors. A two-state space division is established (e.g. inside or outside of an ellipsoid), and the vertex is moved into the correct subspace according to the watermark bit to be hidden. Ohbuchi et al. [OMA98] described the "Tetrahedral Volume Ratio Embedding" algorithm that is invariant to *affine transformation* (note that the affine transformation is different from the similarity transformation in sense that the former also includes shears).

Song and Cho [SC04] presented an interesting means for easily using the existing image watermarking techniques on 3-D meshes. A bounding cylinder is first generated for the cover mesh, and then a regular sampling is carried out on the profile of this cylinder. For each sample, the authors calculate the horizontal geodesic distance from the sample to the mesh surface and take this value as the brightness of this sample pixel. A watermark can then be inserted in the obtained pseudo-range image by using 2-D image watermarking techniques. The changes on horizontal geodesic distances after this image watermarking have to be reflected on the 3-D mesh shape by modifying the positions of related vertices. At last, Bennour and Dugelay [BD06] proposed to embed watermarks in the 2-D contours of a 3-D mesh object.

To sum up, the main drawback of the robust mesh watermarking techniques in the spatial domain is their relatively weak robustness against connectivity attacks, except for the histogram-based and EGI-based techniques. For blind schemes, the watermark synchronization is a difficult problem, because both the attacks and the embedding process itself (the causality problem) can desynchronize the watermark. However, these methods often have the advantage of a high capacity, and are easy to implement. In addition, we can see that in order to resist connectivity attacks in a blind way, the existing spatial methods select certain kinds of connectivity-invariant features as their watermarking primitives. Such features, to some extent, capture the essential property of the mesh shape, and thus are more or less preserved after connectivity changes. By following this idea, in Chapter 6, we propose a blind and robust scheme that takes the mesh's volume moment as the watermarking primitive. This analytic and continuous moment only relates to the 3-D shape represented by the mesh and is independent from mesh's connectivity information as long as its basic shape is preserved.

3.2.3.2 Robust techniques in transform domain

Most successful robust and blind image watermarking algorithms are based on spectral analysis. A better imperceptibility can be obtained due to the spreading effect of the embedded watermark in all the spatial parts of the cover content, and by taking advantage of the masking effect of the human visual system. A better robustness can be achieved if the watermark is embedded in the low and medium frequency parts. In fact, these components are visually important and the attacks tend to conserve them in order to keep a satisfactory visual quality of the attacked content. But unfortunately, for 3-D meshes, there does not exist yet a spectral analysis tool that is efficient and robust enough. Indeed, the irregular sampling nature of 3-D meshes makes spectral analysis

difficult. Almost all the existing mesh frequency analysis tools have their limitations. Besides the algorithms that embed watermarks in the spectral coefficients obtained by direct frequency analysis of the cover mesh, here we also present the algorithms based on multiresolution analysis. The basic idea behind both kinds of watermarking methods is the same: the watermark is embedded through modulation of the data obtained after a certain mesh transformation.

Robust techniques in transform domain based on direct frequency analysis

Researchers have tried different kinds of mesh frequency analysis tools for robust watermarking.

In the conventional combinatorial Laplacian spectral analysis of 3-D meshes, we first construct a symmetric Laplacian matrix D of dimension $N_V \times N_V$ (N_V being the number of mesh vertices) purely depending on the mesh connectivity. If the vertices v_i and v_j are connected by an edge, then the elements d_{ij} and d_{ji} of the matrix D are set to be -1; otherwise, they are set to be 0. Each diagonal element d_{ii} is equal to the valence of the vertex v_i. We carry out eigen-decomposition of this matrix and obtain its N_V eigenvectors and N_V eigenvalues. Then, the eigenvectors are normalized so as to have unit norms, and are also sorted in an ascending order according to their associated eigenvalues, i.e. their associated frequencies (actually, the associated frequency is the square root of the corresponding eigenvalue [DF88]). The N_V-sized spectral vectors $\tilde{\mathbf{x}} = (\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_{N_V})$, $\tilde{\mathbf{y}} = (\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_{N_V})$, $\tilde{\mathbf{z}} = (\tilde{z}_1, \tilde{z}_2, ..., \tilde{z}_{N_V})$ are calculated respectively as the projections of the three mesh vertex coordinate vectors $\mathbf{x} = (x_1, x_2, ..., x_{N_V})$, $\mathbf{y} = (y_1, y_2, ..., y_{N_V})$, $\mathbf{z} = (z_1, z_2, ..., z_{N_V})$ on the N_V normalized and sorted eigenvectors. The k-th spectral amplitude is normally defined as $c_k = \sqrt{\tilde{x}_k^2 + \tilde{y}_k^2 + \tilde{z}_k^2}$. Figure 3.8 illustrates the spectral amplitudes of the simplified Bunny mesh that has 100 vertices.

This combinatorial mesh spectral analysis was originally derived within the framework of graph theory [Big93], and then used by Karni and Gotsman [KG00] for mesh compression. It was later introduced by Ohbuchi et al. [OMT02] for robust mesh watermarking. In their method, a multi-bit non-blind watermark is embedded in the cover mesh through an additive modulation of its low and medium frequency coefficients. Equation (3.1) describes the modulation scheme of the spectral coefficient $\tilde{x}_k$ (resp. $\tilde{y}_k$, $\tilde{z}_k$, with the same embedded bit), where $w_k \in \{-1, +1\}$ is the watermark bit to be inserted, $a_k \in \{-1, +1\}$ is a pseudo-random bit generated by using a secret key, and α is the watermark embedding strength.

$$\tilde{x}_k^{'} = \tilde{x}_k + w_k a_k \alpha. \tag{3.1}$$

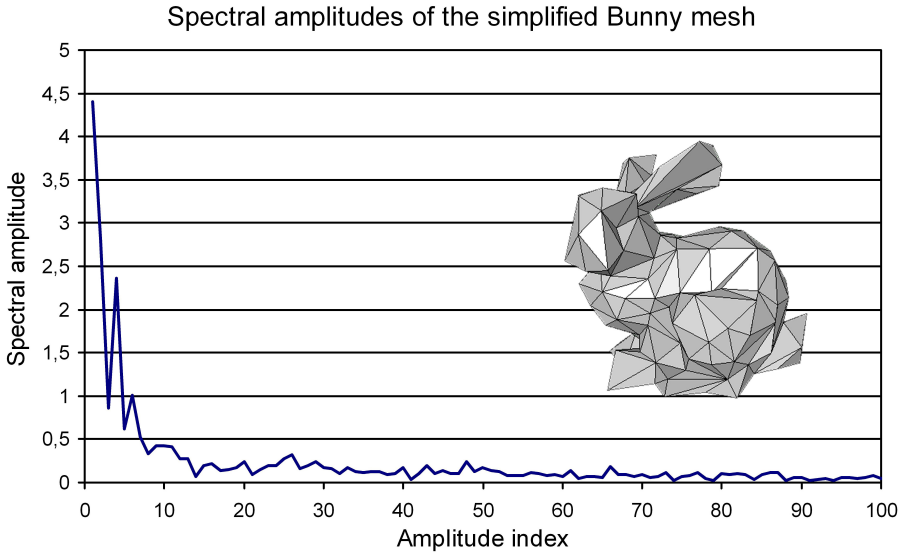


Figure 3.8: The combinatorial Laplacian spectral amplitudes of the simplified Bunny mesh that has 100 vertices.

In order to extract a watermark bit, we first compute a mean value q_k as given by Equation (3.2), where $\tilde{x}_k^{'}$, $\tilde{y}_k^{'}$, $\tilde{z}_k^{'}$ are the spectral coefficients of the input mesh of the watermark extraction algorithm (it is quite likely that this input mesh is an attacked stego model).

$$q_k = \frac{1}{3}\left[\left(\tilde{x}_k^{'} - \tilde{x}_k\right).a_k + \left(\tilde{y}_k^{'} - \tilde{y}_k\right).a_k + \left(\tilde{z}_k^{'} - \tilde{z}_k\right).a_k\right]. \tag{3.2}$$

It is easy to deduce that if there is no attack on the watermarked mesh, we will have $q_k = \alpha.w_k$. Therefore, we simply take the sign of q_k as the extracted watermark bit $\hat{w}_k$, i.e. $\hat{w}_k = sign(q_k)$. The combinatorial Laplacian spectral analysis can also be used to watermark a cloud of points. In that case, it is necessary to construct a fictive connectivity on the cloud of points before the combinatorial Laplacian spectral analysis and the watermark embedding. Cotting et al. [CWPG04] and Ohbuchi et al. [OMT04] proposed different fictive connectivity construction methods.

In general, there exist two serious problems for this combinatorial Laplacian spectral analysis, in the context of robust mesh watermarking. First, the computation time increases rapidly with the mesh complexity due to the diagonalization of the $N_V \times N_V$ Laplacian matrix during the derivation of the spectral analysis bases, i.e. the matrix eigen-vectors. In order to resolve this computation complexity problem, before watermark embedding and extraction, researchers often segment the input mesh into several patches, each of which possesses fewer vertices. Then, local spectral analysis, which requires much less computation, is carried out on these patches; after that, the watermark

is embedded in or extracted from the obtained local patch spectral coefficients. The computation complexity issue can also be alleviated by carrying out a partial derivation (instead of a complete derivation) of the solution to the established eigen-problem. For example, in [OMT02], the authors only deduce a limited number of low-frequency eigenvalue and eigen-vector pairs by using the Arnoldi method [GGvL96]. These computed eigenvalues and eigen-vectors will later be used during the watermark embedding or extraction. The second problem is that the spectral analysis bases depend entirely on the mesh connectivity information and thus the obtained spectral coefficients, which often constitute the watermarking primitives, are not robust against connectivity attacks on the watermarked mesh (c.f. Section 7.2.2 for some experimental results). In order to overcome this fragility, before extracting watermark from a certain mesh, it is necessary to perform a preprocessing step of resampling so as to recover exactly the same connectivity as that of the cover mesh. Due to these two problems, most of the existing robust mesh watermarking schemes based on the combinatorial Laplacian spectral analysis are non-blind, such as the techniques described in [OMT02], [LDD07], [ABB07] and [ABB08]. The watermark extraction in these techniques needs the original cover mesh to recover the same mesh connectivity and/or to ensure the same mesh patching.

Blindness in the mesh spectral domain was first exploited by Cayre et al. [CRAS*03]. Their watermark embedding scheme is a substitutive one. As illustrated in Figure 3.9, the median value of the set $\{\tilde{x}_k, \tilde{y}_k, \tilde{z}_k\}$ is substituted according to the bit to be embedded and to the quantization intervals established by the maximum and minimum values of the same set. The watermark modulation scheme in their method is blind; however, in order to ensure a satisfying robustness (especially against connectivity attacks), it seems still necessary to have the original cover mesh available at the watermark extraction phase.

Two recently proposed blind spectral methods [LB08, LPG08] have achieved a better robustness, especially against connectivity attacks. In the method of Luo and Bors [LB08], the robustness relies on the stability of the distribution of a group of high-frequency coefficients obtained after the combinatorial Laplacian spectral analysis presented above. This method needs to compute the whole spectrum of the cover mesh; therefore, it is not applicable on large meshes having more than 10000 vertices because the calculation of the whole spectrum for such meshes is extremely time-consuming. Lately, Luo et al. [LWBL09] improved this point by introducing a robust and "blind" preprocessing of mesh registration and segmentation before the watermark embedding/extraction. Note that at extraction, this "blind" preprocessing does not need the original non-watermarked mesh. In this way, the cover mesh and the (attacked) stego

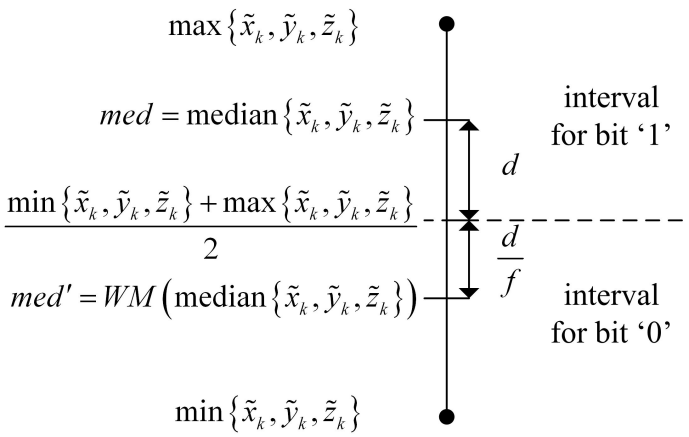


Figure 3.9: The substitutive modulation of the spectral coefficient in the algorithm of Cayre et al. [CRAS*03]. The original median value *med* is substituted by the watermarked value med'. The lower interval stands for the bit '0', and the upper interval represents the bit '1'. Here, the watermark bit to be embedded is '0', so the median value is moved from the upper interval to the lower interval. If the original difference between *med* and the average value of $\max\{\tilde{x}_k, \tilde{y}_k, \tilde{z}_k\}$ and $\min\{\tilde{x}_k, \tilde{y}_k, \tilde{z}_k\}$ is d, then after bit embedding, the difference between med' and the same average value is $\frac{d}{f}$, where f is a parameter that controls the trade-off between watermark robustness and imperceptibility.

mesh are robustly and consistently split into several small patches on which the spectral analysis requires much less computation. Unlike the other spectral schemes, the method of Liu et al. [LPG08] makes use of a new mesh spectral decomposition tool, namely the manifold harmonics analysis [VL07, VL08] (more details about this analysis will be presented in Chapter 7). The spectral decomposition bases in the manifold harmonics analysis are related to both geometry and connectivity of the mesh. Experimentally, the obtained spectrum coefficients are quite robust, even after connectivity alterations (c.f. Section 7.2.2 for some experimental results). In the method of Liu et al., the low frequency part of the mesh spectrum is split into 5 slots. Then in each slot, one bit is embedded by modifying the relative relationship between a certain selected spectral amplitude and the average of the other spectral amplitudes in the same slot, in a very similar way as in the method of Cayre et al. [CRAS*03]. The main drawback of the scheme of Liu et al. is its low capacity (5 bits). Finally, it is worthwhile pointing out that the main difficulty encountered while using manifold harmonics analysis for mesh watermarking is the occurrence of the causality problem, which results in the fact that we cannot easily ensure the correctness of the watermark embedding. The solving of this causality problem often leads to an extra computation time and a decrease of the watermarking capacity. The details of this causality problem, as well as a corresponding solution will be presented in Chapter 7. In that chapter, we will propose an improved ro-

bust and blind multi-bit mesh watermarking method based on the manifold harmonics analysis.

Some other direct mesh frequency analysis tools have been used for robust watermarking. Wu and Kobbelt [WK05] reported a spectral method that is based on the radial basis functions. Murotani and Sugihara [MS03] proposed to embed the watermark by using the mesh singular spectral analysis. In both methods, the spectral analysis bases are entirely dependent on the mesh geometry information. The derivation of these geometry-related spectral bases is made computationally efficient, because the matrix that is to be diagonalized has a very low dimension $Dim << N_V$. However, it seems that in both watermarking algorithms, the derived spectral bases are not quite robust; accordingly, the algorithms remain non-blind in order to ensure a satisfactory robustness against various attacks.

In all, although current 3-D mesh spectral analysis tools seem not efficient or robust enough, they provide a promising domain in which we may directly apply the basic idea of the existing image spectral watermarking methods.

Robust techniques in transform domain based on multiresolution analysis

Besides the direct mesh spectral domain, a robust watermark can also be embedded in the mesh multiresolution domain.

Based on the regular wavelet analysis presented in the subsection concerning the fragile watermarking techniques in transform domain (i.e. Section 3.2.1.2), Kanai et al. [KDK98] proposed a non-blind algorithm that modifies the ratio between a wavelet coefficient norm and the length of its associated edge, which is invariant to similarity transformations. Uccheddu et al. [UCB04] described a correlation-based blind detectable (1-bit) watermarking algorithm for semi-regular meshes. In their method, the cover semi-regular mesh is first normalized to a canonical spatial pose, then the watermark signal is embedded through additive modulation of the norms of the wavelet coefficient vectors at a certain appropriate resolution level. In Chapter 5, we will present a robust and blind multi-bit watermarking method for semi-regular meshes, also based on the wavelet transform.

With a remeshing [AUGA08] step before wavelet decomposition, the regular wavelet analysis can be extended to irregular meshes. Jin et al. [JDBP04] used such a technique to insert a non-blind watermark into both the coarsest representation and the spherical wavelet [SS95] coefficients of an irregular mesh. However, this remeshing step seems not robust enough and can also introduce non-negligible noise into the mesh model.

Consequently, the watermark robustness and imperceptibility may be degraded due to this remeshing preprocessing. Using a direct irregular mesh wavelet analysis tool without any assisting remeshing step [VP04], Kim et al. [KVJP05] devised a blind algorithm; however their method is fragile to connectivity attacks.

Other multiresolution analysis tools have also been employed to develop 3-D mesh watermarking algorithms. Hoppe [Hop96] presented a multiresolution decomposition method based on the iterative edge collapse operation. The dual reconstruction procedure is based on the iterative vertex split operation. Figure 3.10 illustrates these two operations. Praun et al. [PHF99] proposed a robust and non-blind mesh watermarking scheme that makes use of these multiresolution decomposition and reconstruction operations. They pick out the vertex split steps of the reconstruction process that introduce the most significant geometric modifications. For each vertex to be split in these selected steps, the authors define a zone containing all its incident facets in the coarser-level mesh. They then find the corresponding area in the original dense mesh and take this area as the watermark bit carrier. One bit is embedded in each area by deforming it using a well-designed modulation function so as to ensure the watermarking imperceptibility. In fact, their watermarking technique lies between the conventional spatial methods and the classical multiresolution methods. Here, the multiresolution analysis is utilized to find the salient spatial parts of the mesh, and the watermark insertion in these parts is supposed to be more robust. Unfortunately, these iterative edge collapse operations are dependent on the mesh connectivity. Therefore, this algorithm is non-blind mainly due to the connectivity recovery before watermark extraction. At last, Yin et al. [YPSZ01] embed a robust, but non-blind watermark (connectivity recovery is also necessary) in the coarsest representation after a mesh multiresolution analysis based on the Burt-Adelson pyramid decomposition [GSS99].

In all, quite similar to the existing direct spectral analysis tools, the current mesh multiresolution analysis schemes have either connectivity restrictions or robustness deficiencies (especially against connectivity attacks). For the majority of the watermarking techniques in the multiresolution domain, registration and resampling are necessary so as to ensure a satisfactory robustness; but these preprocessing steps inevitably make the techniques non-blind.

Besides the direct spectral analysis and the multiresolution analysis, mesh parameterization [FH05] has also been used for watermarking. Parameterization is a technique that maps a 3-D mesh onto a bidimensional domain, and thus probably enables the direct use of existing 2-D image watermarking algorithms on 3-D meshes. Li et al.

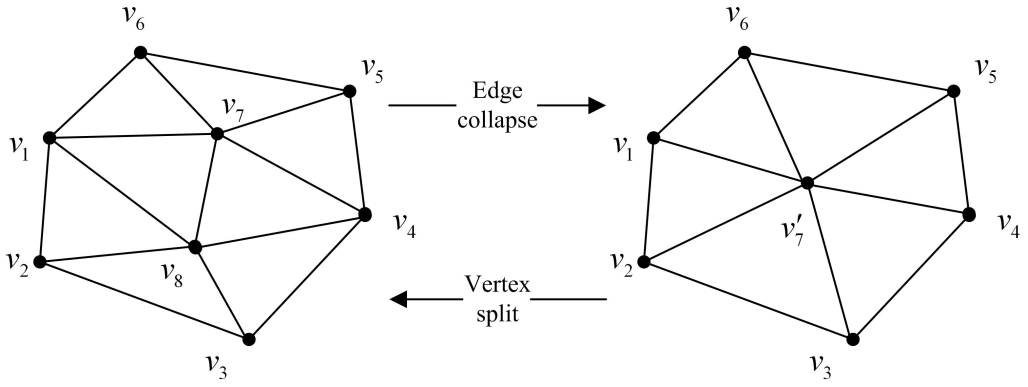


Figure 3.10: The edge collapse and vertex split operations used for mesh multiresolution analysis and synthesis [Hop96]. In the edge collapse operation, the edge connecting v_7 and v_8 is removed, and a new vertex $v_7^{'}$, whose coordinates are calculated according to a certain rule, is connected to all the vertices having been incident to v_7 and v_8; while in the vertex split operation, the vertex $v_7^{'}$ is split into v_7 and v_8, and with the information saved during the edge collapse operation, we can exactly recover the original local geometry and connectivity.

[LZP*04] first convert the geometry of the cover mesh into evenly sampled 2-D spherical signals by using the spherical parameterization [GGS03, PH03]. Then, the watermark is embedded in the spherical harmonic coefficients of the obtained 2-D spherical signals. This algorithm is semi-blind since it needs the spherical parameterization information of the original non-watermarked mesh at extraction to ensure a good robustness.

3.3 Attack-Centric Investigation

The attacks constitute a critical factor when designing 3-D mesh watermarking algorithms. In this section, we will discuss the different kinds of attacks on watermarked meshes and present the existing solutions in order to obtain the robustness against them.

3.3.1 Robustness against geometry attacks

This kind of attack only modifies the geometry part of the watermarked mesh, i.e. the vertex coordinates. The vertex number and the adjacency relationship between vertices are always kept unchanged.

3.3.1.1 Similarity transformation

In most cases, similarity transformation is considered as a common geometry operation on a 3-D mesh rather than a malicious attack, against which even a fragile or a high-capacity watermark should be able to resist. It includes translation, rotation, uniform

scaling and the combination of the above three operations. Figure 3.11.(b) illustrates a Rabbit model that has been subject to a similarity transformation. In general, there are three different strategies to build a watermark that is immune to this routine operation.

The *first solution* is to use watermarking primitives that are invariant to similarity transformations. Ohbuchi et al. [OMA97] provided a list of such primitives. The most utilized one is the ratio between two measures of a triangle (e.g. height or edge length). Some primitives used in the existing blind spatial techniques are also invariant to similarity transformations, such as the quantized position of the projection of a vertex on its opposite edge in a triangle [CM03, WC05, CW06], and the relative position of a vertex to a zone defined by its 1-ring neighbors [HB02, Boro6]. These primitives are all some relative geometric measures between several absolute and individual ones.

Not only the watermarking primitives, but also the synchronization schemes have to be insensitive to similarity transformation. Existing synchronization mechanisms often consist of 1) criteria for choosing the first watermarking primitive and 2) further spreading schemes. For example, in the method of Cayre and Macq [CM03], each triangular facet is considered as a two-state object with one entry edge and two exit edges. The authors take the longest edge in a certain facet intersecting with the mesh's most significant principal axis as the first entry edge. The spreading scheme is determined by a secret key: if the next bit in this key is '0', then the first edge in the clockwise direction from the entry edge inside the current facet is chosen as the next entry edge and the next triangular facet is thus determined; and *vice versa*. In the method of Bors [Boro6], a reference vertex is first selected as the one having the smallest average length of the vertex's incident edges, and then the other vertices are ordered according to their distances to this selected reference vertex. The causality problem arises in the method of Bors because after watermark embedding, the order of the vertices may have been changed. In order to resolve this problem, the author introduces a post-processing step to recover the original vertex orders. Another option is the so-called indexing watermarking scheme. One example is given in [OMA97]. A group of four triangles are combined together as a watermarking primitive. One of the four triangles is modified to indicate the existence of watermark bits in this macro-group. Two other triangles are used to hide the watermark bits. The index of these bits in the entire watermark sequence is hidden in the last triangle. The advantage of this option is that the extraction failure of a certain bit (or certain bits) will not influence the extraction (with correct indices) of the posterior bits. However, the adoption of the indexing scheme inevitably decreases the watermarking capacity since some primitives have to be used for the embedding of the watermark bit

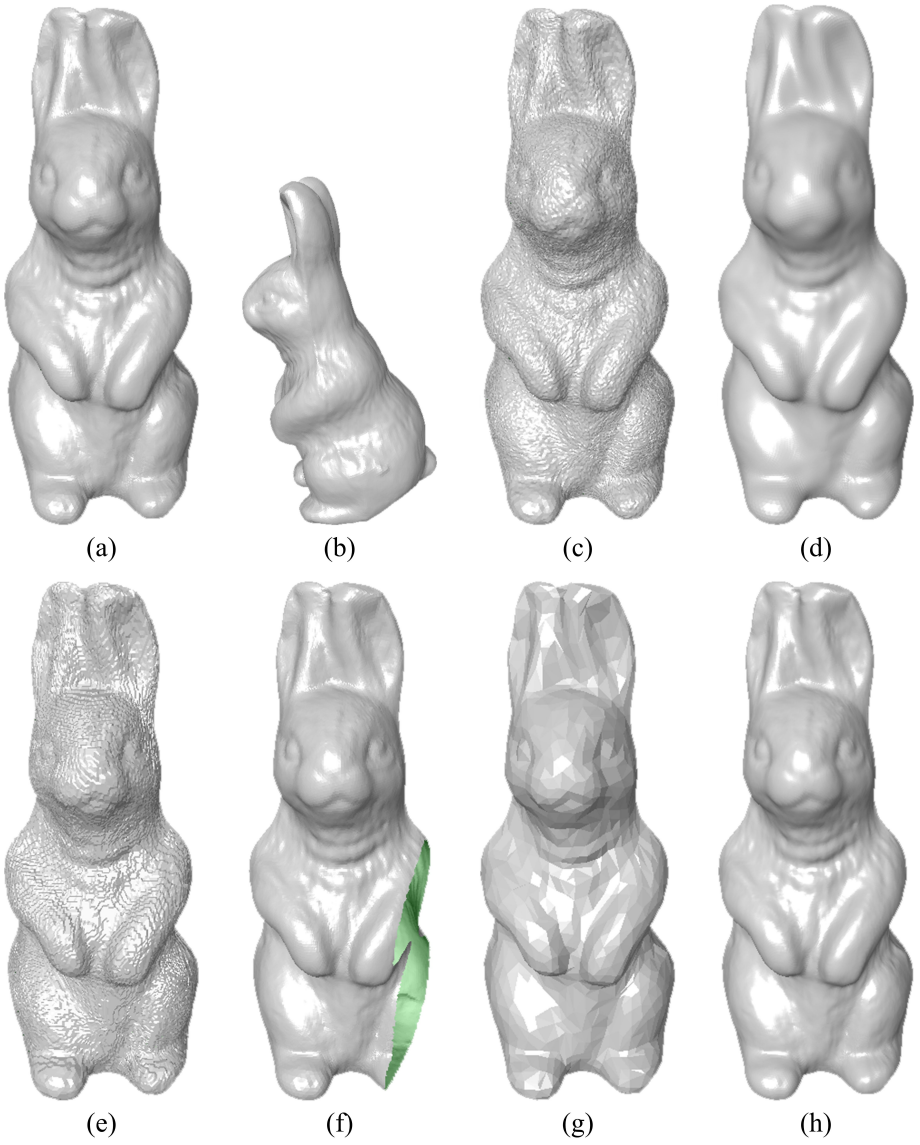


Figure 3.11: The original Rabbit model and seven attacked versions: (a) the original one; (b) after similarity transformation; (c) after random noise addition; (d) after smoothing; (e) after vertex coordinate quantization; (f) after cropping; (g) after simplification; (h) after subdivision.

indices.

The invariance to similarity transformation can also be achieved in the regular wavelet domain by watermarking the ratio between the norm of a wavelet coefficient vector and the length of its support edge [KDK98, CLLP05]. Moreover, if we expect robustness against the general affine transformation, the Nielson-Foley norm [NF89] can be a good candidate of the watermarking primitive. Benedens and Busch [BB00] quantize this norm, and Wagner [Wag00] replaces some medium-important bits of this norm to embed affine-transformation-invariant mesh watermarks.

The *second solution* is to embed watermarks in an invariant space to the similarity transformation. One such space can be obtained by carrying out the following steps [KTP03]:

1. Calculate the center of the mesh and translate the model so that its center coincides with the origin of the objective Cartesian coordinate system;
2. Carry out a uniform scaling so that the whole mesh is bounded within a unit sphere;
3. Calculate the principal axes of the mesh and rotate the object so that its principal axes coincide with the axes of the objective Cartesian coordinate system.

The watermark is then embedded in this new space. Unfortunately, the causality problem may occur in such watermarking schemes because the variables used in the above steps, such as the mesh center and the mesh principle axes would be possibly changed after watermark embedding. Consequently, at the watermark extraction stage, we may have some extent of error when reconstructing this canonical watermark embedding space in a blind way. In order to avoid this error, it would be necessary to at least memorize some feature values of the embedding space and transmit them to the extraction side, but this supplementary information transmission would make the technique semi-blind, or even non-blind. Note that not all watermark embedding schemes need all of the above three steps: which steps are required depends on the nature of the watermarking primitive.

The *third solution* is to carry out a registration of the input mesh at extraction so as to recover the same spatial pose of the original cover mesh. Low-precision registration methods use singular spectral coefficients [MS03], eigenvectors of the vertex correlation matrix [OTMM01], inertial moments [OMT02], or characteristic points [WK05] of the two meshes. High-precision methods often need user interactions to determine a good initial condition, and then the registration is realized by iteratively minimizing a sum

of local errors [YPSZ01, OMT02, ABB08]. This third solution will obviously make the algorithms non-blind, but meanwhile provides a better robustness.

3.3.1.2 Signal processing attacks

A mesh can be considered as a signal in three-dimensional space. There are counterparts of the traditional one-dimensional signal processing techniques for 3-D meshes, such as random noise addition, smoothing, enhancement and geometry compression (usually realized by vertex coordinate quantization). Figures 3.11.(c)-(e) illustrate three examples of signal processing attacks. Although these operations can be very harmful to the embedded watermark, they are really common manipulations in many applications such as the video games and the cartoon film making.

Random noise addition, smoothing and enhancement can be modeled in the spectral domain by a modification of the high-frequency components. Vertex coordinate quantization can be thought as an addition of a certain kind of noise, but its effect is somewhat complicated. In general, the transform-domain-based techniques that modify the low and medium frequency parts are more robust against these attacks than the other techniques. Note that for the additive watermarking schemes that embed the watermark by modulating (i.e. perturbing) spectral coefficients obtained after a direct mesh frequency analysis, embedding in the low frequency part is both more robust and more imperceptible compared to that in the high frequency part if they have the same embedding intensity [SCOT03, ZvKD07]. Different additive modulation schemes have been developed. Ohbuchi et al. [OTMM01] proposed to repeat the watermark bit embedding in the low and medium frequency coefficients with constant intensity. Wu and Kobbelt [WK05] watermarked only the very low frequency coefficients and proposed an adaptive embedding intensity that is proportional to the amplitude of the spectral coefficient. Lavoué et al. [LDD07] presented another modulation scheme, in which the intensity is linear for the low and medium frequency coefficients and constant for the high frequency coefficients.

Spatial techniques seem less robust against signal processing attacks. One exception is the histogram-based techniques [ZTP05, CPJ07]. Statistical mesh shape features used in these techniques tend to be preserved after such attacks because they represent global features of groups of mesh combinatorial elements. Another efficient solution is to search for an adaptive watermark embedding intensity based on a local geometric analysis of the mesh shape. This analysis can be based on the average length of the incident edges of a vertex [AE03], the geometric distortion induced by a vertex split op-

eration [PHF99], the minimal incident edge length of a vertex [YPSZ01], or the possible normal direction variance of the incident facets of a vertex after watermark embedding [YIK03]. The basic idea is to increase the watermarking intensity at the locations where the induced distortion is more imperceptible. At last, redundant insertion [OTMM01] and the use of error correction code [LDD07] can sometimes significantly reinforce the robustness against these attacks.

3.3.1.3 Local deformation

A local deformation can be completely imperceptible if we do not have the original mesh for comparison, but it can seriously disturb the embedded watermark, especially its synchronization mechanism.

One natural solution is to divide the mesh into several patches and repeat the watermark insertion in each patch. This decomposition can be based on surface curvature, high-level shape analysis of the 3-D model or simply a discretization of the θ and/or φ domain in the spherical coordinate system. As mentioned previously, segmentation into patches may also efficiently decrease the watermark embedding time for some transform-domain-based watermarking techniques, such as the methods described in [OMT02], [CRAS*03], [ABB07], [ABB08] and [LWBL09]. At extraction, one has to realize exactly the same decomposition. This is relatively simple and robust for non-blind techniques due to the availability of the cover mesh or the non-attacked stego mesh at extraction. On the contrary, devising a blind algorithm capable of resisting local deformation is a very difficult task. The segmentation or discretization methods will probably fail at the extraction phase because the key parameters, such as the surface curvature, the mesh center or the mesh principal axes, will certainly be disturbed after the watermark embedding itself (the causality problem) or an attack (e.g. a local deformation). This issue forces the researchers to devise mesh segmentation schemes that are robust against various attacks, especially the local deformation and the cropping (c.f. Section 3.3.2). Rondao-Alface et al. [RAMC07] have made some efforts in this direction. They proposed a segmentation scheme that is based on mesh feature points. These feature points are obtained by geodesic distance analysis of the mesh surface and are relatively robust against local deformation and cropping. However, it seems that the robustness of their blind watermarking algorithm still needs some improvement. Another solution to resisting local deformation is the indexing watermarking schemes, which were mentioned in Section 3.3.1.1. However, it seems quite difficult to devise a blind indexing watermarking scheme that can withstand the connectivity attacks.

3.3.2 Robustness against connectivity attacks

Connectivity attacks mainly consist of cropping, surface simplification, subdivision and remeshing. In these attacks, the initial combinatorial elements (vertices, edges and facets) of the watermarked mesh may be removed while some new elements can be inserted. In general, connectivity attacks are much more difficult to handle than the geometry attacks. Actually, there exist very few blind mesh watermarking schemes which can resist connectivity attacks.

Cropping is a special attack (c.f. Figure 3.11.(f) for an example), and some researchers prefer to regard it as a geometry attack because its consequence is quite similar to that caused by a strong local deformation. Watermark repetition in different patches and indexing watermarking schemes seem the most efficient ways in order to resist cropping.

Concerning the other connectivity attacks (Figure 3.11.(g) and 3.11.(h) illustrate respectively the simplified and the subdivided Rabbit), the algorithms that take the average normal directions of groups of facets as the primitives [Ben99a, KKL*03, LLKL05, LK07], or the histogram-based algorithms [ZTP05, CPJ07], are less sensitive. The watermarking primitives used in these methods approximately describe the global shape of the mesh and thus are more or less conserved after connectivity alterations. Note that although the above histogram-based techniques are quite robust against isotropic connectivity modifications, they remain vulnerable to anisotropic connectivity attacks. Indeed, under anisotropic connectivity alterations, the mesh center and the mesh principle axes calculated in these methods, as well as the distribution of the used histograms, will be seriously disturbed. Consequently, the watermark extraction may fail because of the failures in recovering the watermark embedding space and in extracting watermark bits from the disturbed primitives. Other spatial techniques are less robust due to the geometric change of the watermarking primitives and also to the desynchronization problem. Meanwhile, the mesh spectral analysis tools used in most of the existing spectral mesh watermarking methods are not robust [OMT02, CRAS*03, MS03, WK05], especially under connectivity attacks. The spectral coefficients obtained by the manifold harmonics transform are quite robust to connectivity changes, but the watermarking scheme using this transform suffers from the causality problem [LPG08]. The consequences are that the embedding time will be increased and that the watermarking capacity may be greatly decreased. The existing multiresolution analysis tools either have connectivity restrictions, or are not robust enough against connectivity changes. Hence, to ensure a satisfactory robustness for these methods, the authors usually recommend to perform connectivity restoration before watermark extraction. This restora-

tion procedure can be considered as a resampling of the extraction input mesh (objective mesh) so as to obtain the same connectivity configuration as the cover mesh [YPSZ01, OMT02, YIK03, ABB08] or the non-attacked stego mesh [WK05] (reference mesh). The task is to find, for each vertex in the reference mesh, a corresponding point on the surface of the objective mesh. This correspondence can be established by the nearest neighbor criterion [WK05], ray intersection [YIK03, OMT02, ABB08], or an iterative process targeting to minimize a particular cost function [YPSZ01]. This resampling preprocessing ensures a good robustness against connectivity attacks but inevitably makes the watermarking method non-blind.

3.3.3 Robustness against other attacks

There are mainly two kinds of attacks in this category: *file attack* and *representation attack*. The file attack simply consists in reordering the vertices and/or the facets in the mesh description file. In order to be invariant to this attack, one just needs to make the watermark synchronization scheme independent of the combinatorial element orders represented in the mesh file.

The representation conversion may be the most destructive attack to 3-D mesh watermarks, because after such an attack, the mesh itself will no longer exist (for example, an approximation of a watermarked mesh with an NURBS model or with voxels). Until now, no researcher has mentioned the robustness against the representation attack. In Chapter 6, we present a volume-moment-based scheme that achieves a satisfactory robustness against voxelization.

3.3.4 Comparison between different robust techniques

In Section 3.2.3, we presented the existing robust mesh watermarking techniques by paying more attention on the watermarking primitives. In this section, the routine attacks that aim to remove the embedded robust watermark were classified and the countermeasures to each kind of attacks were discussed. Now, we summarize and compare some typical robust techniques in terms of some widely used evaluation metrics. Tables 3.3 and 3.4 present the comparison results. The values in the column "Embedded bits" are the ones reported in the original papers. Most robustness performances are evaluated qualitatively by a sign ranging from "$--$", which means the least robust, to "$++$", which stands for the most robust. In these two tables, the algorithms are classified according to the embedding domain combined with the watermarking primitives. The categories are: spatial techniques on vertices, spatial techniques on facets,

transform-domain-based techniques using direct spectral analysis, transform-domain-based techniques using multiresolution analysis, and other techniques. In the last group "Other techniques", we list two other representative algorithms (the first one works in the spatial domain and the second one operates in a transform domain), which do not belong to any of the other four classes.

3.4 Conclusion

3-D mesh watermarking is an interesting and promising research area, with many potential applications. However, due to many difficulties stated in Section 3.1, such as the irregularity of the mesh description and the complexity of the possible attacks, the research on 3-D mesh watermarking is still in its early stage, even after ten years of studies of a large community. In this chapter, we have presented a comprehensive survey on 3-D mesh watermarking, with an original attack-centric investigation. From this survey, we can summarize several open problems concerning this research subject. For fragile watermarking, one interesting work would be the design of a scheme that is capable of precisely locating the endured attacks and meanwhile invariant to all the content-preserving operations. For high-capacity watermarking, it is worthwhile to investigate whether it is possible to combine the ideas of geometry-based and order-based methods to construct new schemes that have the good properties of both kinds of techniques. The community also seems very interested in finding new spatial watermarking primitives or new robust spectral-like mesh transformations to construct blind and robust watermarking schemes with better overall performances. Finally, it is very important to establish a benchmark for 3-D mesh watermarking, so as to facilitate the evaluation and comparison of different techniques, and thus to promote the relevant research. We were actually following these future working proposals when carrying out this thesis work. The corresponding results will be presented in the following chapters.

Table 3.3: Comparison between different robust mesh watermarking techniques.

Categories	Algorithms	Embedded bits	Blindness	Local adaptability
Spatial techniques on vertices	Yu et al. [YIK03]	$\approx$ 50 bits	No	Yes
	VFA [Ben99b]	$\approx$ 900 bits	Yes	No
	Zafeiriou et al. [ZTP05]	$\approx$ 20 bits	Yes	No
	Cho et al. [CPJ07]	64 bits	Yes	No
	Bors [Bor06]	$\approx$ 0.2 bits/vertex	Yes	Yes
Spatial techniques on facets	TSQ [OMA97]	$\approx$ 1.2 bits/facet	Yes	No
	Benedens [Ben99a]	$\approx$ 30 bits	Semi	No
	Lee et al. [LK07]	$\approx$ 50 bits	Semi	Yes
Direct spectral analysis techniques	Ohbuchi et al. [OMT02]	32 bits	No	No
	Cayre et al. [CRAS*03]	64 bits	Yes	No
	Wu & Kobbelt [WK05]	24 bits	No	No
	Rondao-Alface & Macq [RAM05]	64 bits	Yes	No
	Liu et al. [LPG08]	5 bits	Yes	No
	Luo et al. [LWBL09]	64 bits	Yes	No
Multiresolution analysis techniques	Kanai et al. [KDK98]	$\approx$ 620 bytes	No	No
	Uccheddu et al. [UCB04]	1 bit	Yes	No
	Praun et al. [PHF99]	50 bits	No	Yes
	Yin et al. [YPSZ01]	250 bits	No	Yes
Other techniques	Bennour & Dugelay [BD06]	$\approx$ 500 bits	No	No
	Li et al. [LZP*04]	24 bits	Semi	No

Table 3.4: Continuation of Table 3.3: Resistance of different robust mesh watermarking techniques against various attacks.

Algorithms	Similarity transform.	Signal processing attacks	Local deform. & cropping	Connect. attacks	Element reorder.
Yu et al. [YIK03]	Regis.	+	−	Resamp.	Invariant
VFA [Ben99b]	+	−	−	−	Invariant
Zafeiriou et al. [ZTP05]	+	+	−	+	Invariant
Cho et al. [CPJ07]	+	+	−	+	Invariant
Bors [Bor06]	++	−	−	−−	Invariant
TSQ [OMA97]	++	−	+	−−	Invariant
Benedens [Ben99a]	Regis.	+	−	+	Invariant
Lee et al. [LK07]	Regis.	+	−	+	Invariant
Ohbuchi et al. [OMT02]	Regis.	++	++	Resamp.	Invariant
Cayre et al. [CRAS*03]	+	+	++	−−	Invariant
Wu & Kobbelt [WK05]	Regis.	++	++	Resamp.	Resamp.
Rondao-Alface & Macq [RAM05]	+	+	++	+	Invariant
Liu et al. [LPG08]	++	+	−−	+	Invariant
Luo et al. [LWBL09]	++	+	−−	+	Invariant
Kanai et al. [KDK98]	+	−	−	−−	Invariant
Uccheddu et al. [UCB04]	++	+	−	−	Invariant
Praun et al. [PHF99]	Regis.	++	++	Resamp.	Resamp.
Yin et al. [YPSZ01]	Regis.	+	−	Resamp.	Resamp.
Bennour & Dugelay [BD06]	Regis.	+	+	−	Invariant
Li et al. [LZP*04]	+	+	+	Resamp.	Invariant

Remark: In this table, "Regis." is short for "Registration" and "Resamp." is short for "Resampling".

Chapter 4

Scalar Costa Scheme

Contents

This chapter briefly presents the scalar Costa scheme (SCS) [EBTG03], which is a widely used quantization-based data hiding technique for image, audio and video watermarking. Later in this manuscript, we will use SCS for blind watermarking of 3-D meshes. We first introduce the SCS from a historical viewpoint. Then, we describe the watermark embedding and extraction mechanisms adopted in this scheme. After that, the performance of the SCS, in terms of capacity, distortion, robustness and security, is presented. Finally, we discuss some important issues that are encountered when using SCS for mesh watermarking.

4.1 A Brief History: From Low-Bits Modulation to Quantization Index Modulation

During the early stage of digital watermarking research in the 1990s, a watermark is often embedded in a multimedia content by means of a simple substitution of the 1 or 2 least significant bits (LSBs) of a certain host feature value by watermark bits. For instance, the values on the last 2 bit planes of the image pixel luminance can be replaced by 2 watermark bits. Such watermarking algorithms [TNM90, vSTO94, Bar97, MDS97] are often referred as low-bit(s) modulation (LBM) schemes. LBM schemes are quite efficient in terms of watermarking capacity and imperceptibility, but are vulnerable to various intentional or unintentional attacks. For example, a simple flipping of the LSBs (i.e. we change bit '0' to bit '1' and bit '1' to bit '0') of the stego content completely erases the embedded watermark, meanwhile not inducing obvious visual distortions.

Researchers quickly noticed the deficiencies of the simple LBM schemes and proposed to use *spread spectrum* (SS) techniques [Dix84, Fli97] for multimedia watermarking. Basically, a pseudo-random noise sequence, which represents the watermark, is added to a certain feature signal of the cover content (often referred as *host signal*); and the watermark retrieval is based on correlation value calculation. For blind schemes, the correlation is often calculated between the watermarked host signal (which has probably been attacked) and the pseudo-random sequence that we want to search. For non-blind schemes, at the detection stage, one generally computes the correlation of the pseudo-random watermark sequence with the difference between the feature signals of the stego and cover contents. The added pseudo-random sequence can be either host-signal adaptive or non-adaptive. Some typical spread spectrum watermarking schemes are described in [WD96], [BGML96], [CKLS97], [PBBC97], [HG98] and [OP98].

In the following, we briefly present the algorithm of Piva et al. [PBBC97] to illustrate the basic idea of the SS watermarking. In their blind and robust image watermarking scheme, the $N \times N$ cover image I is first subjected to a global DCT transformation. Then, the M-length watermark representing sequence $\{w_1, w_2, ..., w_M\}$, which is of standard normal distribution and generated by using a secret key, is added to the last M DCT coefficients of the cover image. The embedding rule is described by Equation (4.1), where $i \in \{1, 2, ..., M\}$, t_{L+i} represents the initial high-frequency DCT coefficient, t'_{L+i} denotes the corresponding modulated DCT coefficient, and α is a global control parameter that adjusts the watermark embedding strength.

$$t'_{L+i} = t_{L+i} + \alpha \left| t_{L+i} \right| w_i. \tag{4.1}$$

From the above equation, we can see that the first L DCT coefficients of the cover image I are not involved in the modulation, mainly to ensure the watermark imperceptibility. Meanwhile, the embedding is host-signal adaptive since the additive modulation term $\alpha \left| t_{L+i} \right| w_i$ is proportional to the amplitude of the host coefficient t_{L+i}. A new modulated image $I^{'}$ can be obtained after performing an inverse DCT of the modulated coefficients $\left\{ t_1, t_2, ..., t_L, t^{'}_{L+1}, t^{'}_{L+2}, ..., t^{'}_{L+M} \right\}$. The authors also make use of the masking effect of the human visual system to enhance the watermark robustness. In fact, each pixel $p^{''}_{i,j}$ of the final watermarked image $I^{''}$ is a linear combination of the corresponding pixels $p_{i,j}$ and $p^{'}_{i,j}$ in I and $I^{'}$, as given by Equation (4.2), with $\beta_{i,j}$ a locally adaptive weighting factor. The derivation of $\beta_{i,j}$ is based on the sample variance of $p_{i,j}$ in a local $R \times R$ window.

$$p^{''}_{i,j} = p_{i,j} \left(1 - \beta_{i,j} \right) + \beta_{i,j} p^{'}_{i,j} = p_{i,j} + \beta_{i,j} \left(p^{'}_{i,j} - p_{i,j} \right). \tag{4.2}$$

At the watermark detection stage, they first compute a correlation value $Corr$ according to the following equation:

$$Corr = \frac{1}{M} \sum_{i=1}^{M} \hat{w}_i \hat{t}_{L+i}, \tag{4.3}$$

where $\hat{w}_i, i \in \{1, 2, ..., M\}$ is the pseudo-random watermark representing sequence that we want to search, and $\hat{t}_{L+i}, i \in \{1, 2, ..., M\}$ is the DCT coefficients of the watermarked and potentially attacked image $\hat{I}$. Finally, the obtained correlation value is compared with a predefined threshold Thr, in order to decide the presence of the watermark represented by the sequence $\{\hat{w}_1, \hat{w}_2, ..., \hat{w}_M\}$. The authors proposed to use the following threshold value, where $t^{''}_{L+i}, i \in \{1, 2, ..., M\}$ are the DCT coefficients of the watermarked image $I^{''}$:

$$Thr = \frac{\alpha}{3M} \sum_{i=1}^{M} \left| t^{''}_{L+i} \right|. \tag{4.4}$$

The main drawback of the spread spectrum watermarking techniques is that they are host-interference non-rejecting methods. It means that the host signal may effectively constitute an interference to the embedded pseudo-random sequence that represents the watermark. Here, we take the SS watermarking scheme of Piva et al. [PBBC97] (presented above) as an example to explain this host-interference problem. First, suppose that in the embedding procedure we do not carry out the watermark enhancement based on visual masking (i.e. we simply set $\beta_{i,j}$ all equal to 1) and that there is no attack on the watermarked image (i.e. $\hat{t}_{L+i} = t^{''}_{L+i} = t^{'}_{L+i}$). We also assume that we want to detect exactly the same pseudo-random sequence as that was initially embedded (i.e. $\hat{w}_i = w_i$). Accordingly, the correlation value calculation given in Equation (4.3) can be rewritten as Equation (4.5). Indeed, the first term in the second line of Equation (4.5)

is the correlation between the host signal and the embedded sequence. This correlation may actually make a negative contribution to the final correlation value *Corr*, thus could interfere the watermark detection. Therefore, in the worst case, it is possible that after a simple additive modulation as in Equation (4.1), we cannot even ensure that the SS watermark has been correctly embedded.

$$\begin{aligned} Corr &= \frac{1}{M}\sum_{i=1}^{M} w_i \left(t_{L+i} + \alpha \left|t_{L+i}\right| w_i\right) \\ &= \frac{1}{M}\sum_{i=1}^{M} w_i t_{L+i} + \frac{\alpha}{M}\sum_{i=1}^{M} \left|t_{L+i}\right| w_i^2. \end{aligned} \tag{4.5}$$

This host-interference issue often leads to a relatively poor performance in terms of capacity and robustness for the SS watermarking algorithms. In general, the crux, as pointed out by Chen and Wornell [CW01a], is that in SS watermarking, the knowledge of the cover image at the watermark embedding side is not sufficiently exploited and utilized. In order to overcome this drawback, we should use host-interference rejecting watermarking methods, one of which is the quantization-based technique.

The research on quantization-based watermarking is in part inspired and encouraged by the work of Costa that was published in 1983 [Cos83]. Costa studied the capacity of the communication channel depicted in Figure 4.1. In this setting, m is the message index to transmit, $\mathbf{x}$ is an independent and identically-distributed (iid) Gaussian host signal, $\mathbf{w}$ is the encoder output, $\mathbf{n}$ is the channel attack in the form of an additive white Gaussian noise (AWGN), and $\hat{m}$ is the estimated message index derived by the decoder. Costa deduced theoretically that the capacity of this communication channel is:

$$C = \frac{1}{2}\log_2\left(1 + \frac{\sigma_w^2}{\sigma_n^2}\right), \tag{4.6}$$

with σ_w^2 the power limit of the encoder output and σ_n^2 the power of the AWGN attack. Interestingly and surprisingly, this capacity is independent of the statistics of the host signal $\mathbf{x}$ and is the same as that of the communication channel for which the host signal $\mathbf{x}$ is also known to the decoder.

As pointed in [CMM99, SK01, SRA06], there exists a straightforward analogy between the communication channel studied by Costa and a blind watermarking system. Table 4.1, which is extracted from [SRA06], presents the duality between these two frameworks. The watermarking research community has benefited a lot from the work of Costa that provided several important insights. First, Costa has shown that in a communication channel, a side information available to the encoder but not to the

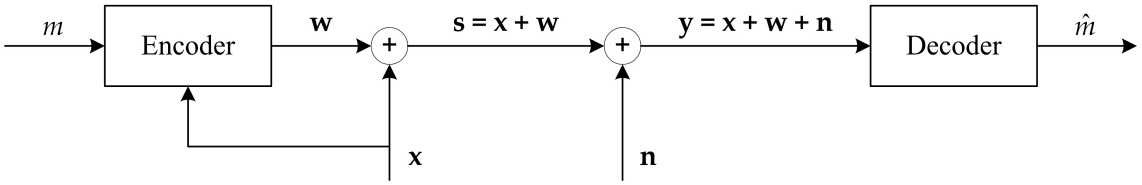


Figure 4.1: The communication channel studied by Costa [Cos83], with a side information **x** available to the encoder but not to the decoder. In this setting, m is the message index to transmit, **x** is an independent and iid Gaussian signal, **w** is the encoder output, **n** is the channel attack in the form of an additive white Gaussian noise, and $\hat{m}$ is the estimated message index derived by the decoder.

Table 4.1: Relationship between communications framework and blind watermarking framework (extracted from [SRA06]).

Communications framework	Blind watermarking framework
Side information	Host signal
Encoder/Decoder	Embedder/Detector(Extractor)
Channel noise	Attacks on stego signal
Power constraints	Perceptual distortion limits
Bandwidth	Embedding signal size (capacity)
Signal-to-noise ratio	Embedding distortion to attack distortion ratio

decoder does not necessarily decrease the transmission rate; in analogy, the blindness of a watermarking scheme does not necessarily result in a performance impairment, at least under certain assumptions. This insight greatly encourages the research on efficient (with respect to capacity and robustness) blind watermarking, which is normally deemed to be a very difficult problem. In addition, Costa also came to the conclusion that instead of trying to cancel the host signal **x** (**x** is actually considered as the first source of interference to the transmitted message index m), the optimal encoder attempts to adapt to it and make use of it through intelligent codebook construction and codeword selection. This conclusion leads to the insight that a good watermark embedding algorithm should exploit the knowledge of the cover content and provides an adaptive and intelligent composition mechanism between the watermark and the cover content.

At the theoretical level, Costa [Cos83] described an optimal encoding method based on a random codebook, which can achieve the capacity limit given by Equation (4.6). Unfortunately, the constructed random codebook contains a very large number of codewords, and therefore the method of Costa is not practical due to computation complexity issues. Researchers proposed several suboptimal but practical coding methods based on quantization of the host signal [CW98, RA99, ESG00, CPGR00]. Historically, quantization (either vectorial or scalar) [GN98] is a widely used technique for lossy compression

of various multimedia contents. Recently, it has also been found very useful in digital watermarking. Actually, the basic idea implied in the LBM schemes can be considered to be the quantization of the watermarking primitive values. However, the constraints existing in these schemes (i.e. only LSBs can be modified) and the lack of security prevents the LBM schemes from becoming effective blind watermarking methods. Hence, researchers started to devise more sophisticated quantization schemes for blind watermarking.

Chen and Wornell [CW01a, CW01b] proposed the Quantization Index Modulation (QIM) data hiding method. QIM is a practical implementation of the ideal Costa coding scheme. In QIM, each watermark message m has an associated quantizer of the host signal $\mathbf{x}$. The quantizer is composed of a number of representing points in the host signal space, which are actually the possible quantized values for $\mathbf{x}$. Figure 4.2 illustrates the basic idea of the binary QIM for which we need to embed one bit in the bidimensional host signal $\mathbf{x}$. In this case, there are two quantizers that are associated respectively to bit '0' and bit '1'. In Figure 4.2, the O's are the representing points of the quantizer corresponding to bit '0', while the $\times$'s are the representing points of the quantizer corresponding to bit '1'. The watermark embedding process consists in replacing $\mathbf{x}$ with the value of the representing point (i.e. quantized value) that is the closest to $\mathbf{x}$ and meanwhile correctly represents the bit to be embedded. Figure 4.2 illustrates an example in which a bit '0' is embedded. The embedding process is equivalent to quantize $\mathbf{x}$ by using the quantizer associated to the watermark bit. At the watermark extraction stage, one simply quantizes the received signal $\hat{\mathbf{x}}$ with all sets of quantizers (for binary QIM there are two quantizers), and takes the watermark message represented by the obtained quantized value as the extraction result. In most cases, the encoder and the decoder adopt the minimum distance criterion when performing the quantization. Correspondingly, the signal space is partitioned into several quantization cells (c.f. Figure 4.2), each of which is associated to a quantized value. The ensemble of these cells constitutes the Voronoi diagram [Aur91] of the signal space associated to the set of quantization points in all the quantizers.

The scalar Costa scheme (SCS) [EBTG03] is a special case of the general QIM method. Indeed, Chen and Wornell studied QIM mainly in the context of high-dimensional vectorial quantization. However, in practical applications, this kind of quantization is not very efficient due to complexity issues. In SCS, one performs scalar quantization for each host signal component x_i (a scalar quantity) and the equivalent vectorial quantizer for the N-length signal $\mathbf{x}$ is in fact the Cartesian product of the N scalar quantizers for the signal components $\{x_1, x_2, ..., x_N\}$. This decomposition effectively simplifies the quanti-

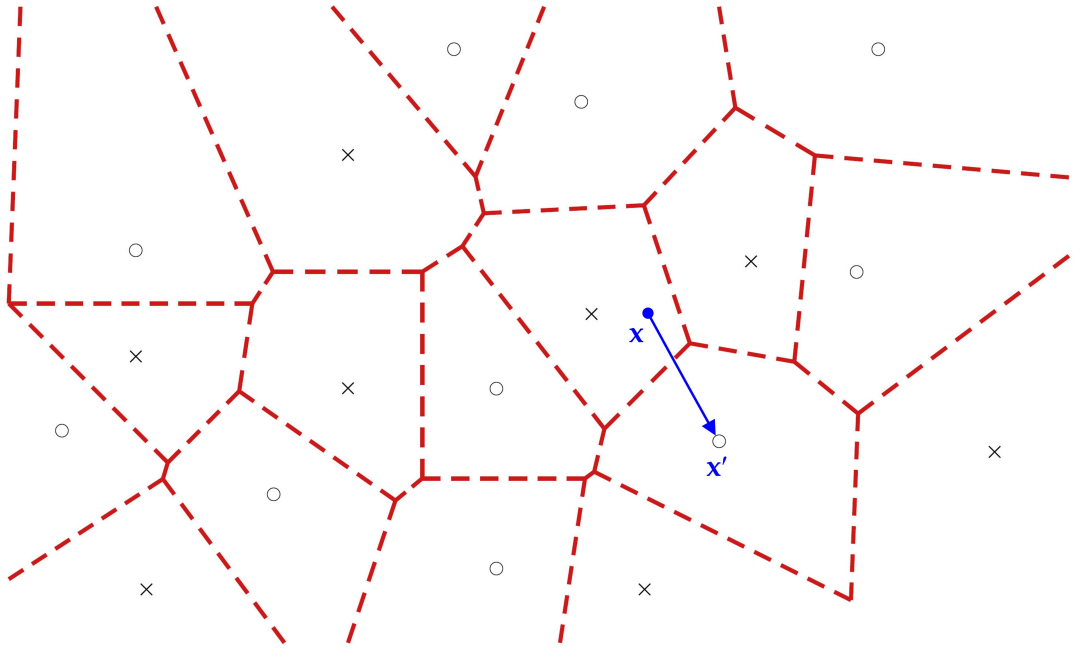


Figure 4.2: This figure, which is extracted from the course notes of Cayre [Cay07], illustrates the basic idea of the QIM watermarking method proposed by Chen and Wornell [CW01a]. In order to embed a bit '0', the host signal $\mathbf{x}$ is substituted by the nearest O value (denoted by $\mathbf{x}^{'}$) in the quantizer associated to bit '0'.

zation procedure, in terms of both algorithm design and practical implementation. In the next section, we will present the embedding and extraction procedures of the SCS.

4.2 Watermark Embedding and Extraction in SCS

Without loss of generality, suppose that we want to hide a sequence of watermark symbols $w_i, i \in \{1, 2, ..., N\}$ in a sequence of scalar quantities $x_i, i \in \{1, 2, ..., N\}$. Each watermark symbol w_i takes its value from alphabet $\mathcal{W} = \{0, 1, ..., R-1\}$ and thus conveys $\log_2 R$ bits. In most cases, we set $R = 2$ so that each symbol w_i conveys one bit; thus, the corresponding watermarking scheme is referred as binary SCS or 2-symbol SCS. In order to carry out the watermark embedding, we first construct a component-wise codebook $\mathcal{U}_{x_i,t_{x_i}}$ for each x_i as follows:

$$\mathcal{U}_{x_i,t_{x_i}} = \bigcup_{l=0}^{R-1} \left\{ u = zS + l\frac{S}{R} + t_{x_i}S \right\}, \tag{4.7}$$

where $z \in \mathbb{Z}$ is an integer, S is the quantization step, $l \in \mathcal{W}$ is the watermark symbol represented by the codeword u, and t_{x_i} is a pseudo-random sequence generated by using a secret key K. As an example, t_{x_i} can be uniformly distributed in $\left[-\frac{1}{2}, \frac{1}{2}\right]$. Initially,

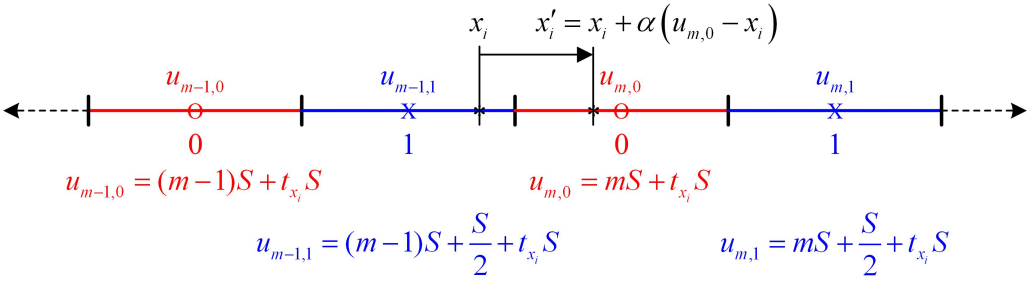
in the context of quantization-based compression, the dither signal t_{x_i} was introduced to realize the de-correlation of the quantization error with the input signal [Sch64, GJ93]. Meanwhile, the actually used (dithered) quantizers are simply the shifted versions of the original non-dithered uniform quantizer; therefore, this quantization scheme also ensures a cheap implementation. In the watermarking context, the dither signal t_{x_i} also serves to randomize the values of the codewords u, with the objective to enhance the watermarking security. In this way, non-authorized watermark extraction and optimal watermark removal can generally be avoided. Note that the codewords in $\mathcal{U}_{d_i,t_{d_i}}$ represent the watermark symbols from $\mathcal{W} = \{0, 1, ..., R-1\}$ in a uniform and interleaved manner.

In order to insert a watermark symbol w_i in x_i, we first find the nearest codeword u_{x_i} to x_i in the codebook that correctly represents w_i. This means that w_i should be equal to value l in the derivation of u_{x_i}, as given by the expression of codeword u in Equation (4.7). Then, the quantized value $x_i^{'}$ is calculated as:

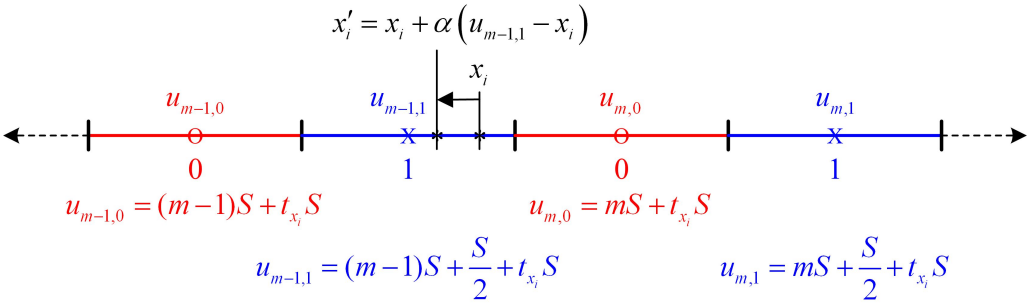
$$x_i^{'} = x_i + \alpha \left(u_{x_i} - x_i \right), \tag{4.8}$$

where α is called the *distortion compensation* (DC) factor which normally takes its value between 0 and 1, i.e. $\alpha \in [0, 1]$. The value of α partially drives the trade-off between induced distortion, robustness and security of the watermarking scheme. However, in practical applications, we should always properly select its value so that it can ensure the correctness of the watermark symbol embedding. This embedding procedure consists in pushing x_i towards u_{x_i}, at least to within the interval $\left(u_{x_i} - \frac{S}{2R}, u_{x_i} + \frac{S}{2R}\right)$, which is the decoding area of u_{x_i} under the minimum distance criterion. In the watermarking literature, the motivation of introducing the distortion compensation mechanism, i.e. the DC factor α, was to achieve optimal performances under different attacks through the adjustment of α. This point will be explained in more details in the next section when analyzing the performance of SCS. Figure 4.3 illustrates the procedure of watermark bit embedding in binary SCS.

In all, the basic idea of the SCS watermark embedding can be summarized as follows: there are multiple codewords in the codebook that represent the same watermark message, and the encoder picks out among them the most appropriate one under a chosen criterion (i.e. the minimum distance criterion) and deduces the watermarked value from this selected codeword (i.e. after distortion compensation post-processing). It can be observed that the SCS embedding process takes the host signal into account, during both codeword selection and watermarked value derivation.



(a) embedded bit is '0'



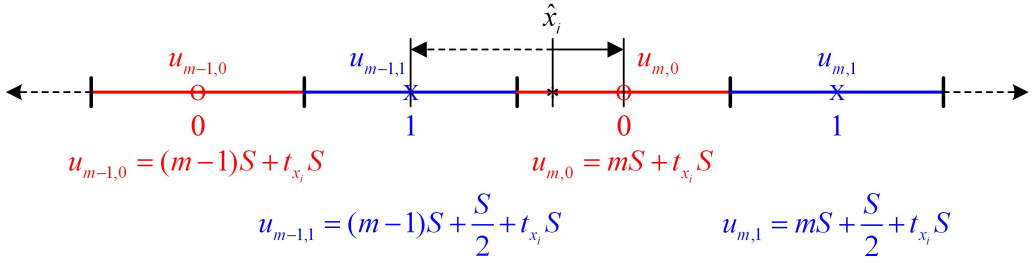
(b) embedded bit is '1'

Figure 4.3: The watermark bit embedding mechanism in binary SCS: (a) the embedded bit is '0'; (b) the embedded bit is '1'. The quantization step S is the distance between two consecutive codewords that represent a same watermark bit (e.g. two consecutive $\times$ points), and m is an integer.

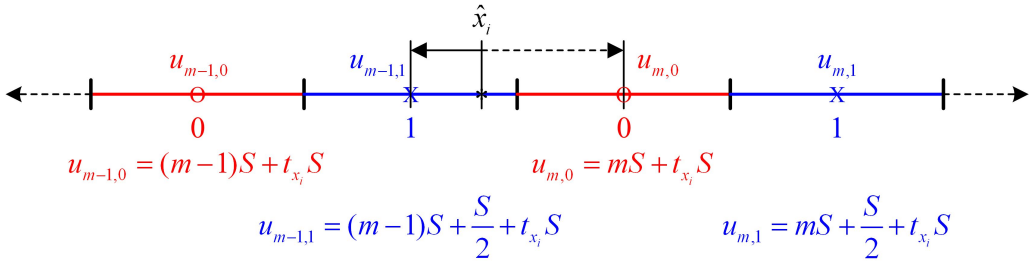
With the knowledge of the secret key K and the values of the parameters R (the watermark symbol number) and S (the quantization step) used during the watermark embedding, the hidden message can be extracted in a totally blind way. First, the same component-wise codebook $\mathcal{U}_{x_i,t_{x_i}}$ is constructed for each received scalar quantity $\hat{x}_i$ from which we would like to carry out the extraction. The watermarking primitive value $\hat{x}_i$ can be different from that in the initial watermarked signal, i.e. $x_i^{'}$, due to the existence of attacks; but this will not impact the codebook construction. Indeed, the values of the comprised codewords depend only on the index of the primitive and are independent from its actual scalar value; thus we can ensure that the same codebook is obtained during extraction as long as the watermarking primitives are correctly synchronized (i.e. indexed). Then, we find in the established codebook $\mathcal{U}_{x_i,t_{x_i}}$ the nearest codeword $u_{\hat{x}_i}$ to $\hat{x}_i$ under the minimum distance criterion as follows:

$$u_{\hat{x}_i} = \arg \min_{u} \|\hat{x}_i - u\|, u \in \mathcal{U}_{x_i,t_{x_i}}. \tag{4.9}$$

The extracted watermark symbol $\hat{w}_i$ is simply the symbol represented by this retrieved codeword $u_{\hat{x}_i}$. Figure 4.4 illustrates the process of watermark bit extraction in binary SCS. The decoding mechanism described above adopts a *hard decision* strategy. If the



(a) extracted bit is '0'



(b) extracted bit is '1'

Figure 4.4: The watermark bit extraction mechanism in binary SCS: (a) the extracted bit is '0'; (b) the extracted bit is '1'. The quantization step S is the distance between two consecutive codewords that represent a same watermark bit (e.g. two consecutive $\times$ points), and m is an integer.

set of all valid watermark sequences is available to the extractor, then a *soft decision* is possible, for instance based on the maximum correlation rule such as in [RA99].

4.3 Discussion on SCS Performance

The main advantages of using SCS for blind watermarking are its easy and low-cost implementation and its high flexibility between different watermark evaluation metrics such as distortion, capacity, robustness and security.

The watermark induced *distortion* of SCS relates to the quantization step and the distortion compensation factor. Indeed, the peek distortion D_{max} for each scalar component x_i is determined as:

$$D_{max} = \alpha \frac{S}{2}. \tag{4.10}$$

It has been theoretically proven [Sch64, GJ93] that by using a uniformly distributed dither signal (e.g. $t_{x_i} \sim U\left(-\frac{1}{2}, \frac{1}{2}\right)$), the quantization error signal $\mathbf{q}$ before the distortion compensation post-processing (for each scalar component x_i, this error is denoted by $q_i = u_{x_i} - x_i$) is statistically independent of the host signal $\mathbf{x}$ that is assumed to have an almost uniform probability density function in a quantization bin. Moreover, the power of the signal $\mathbf{q}$ is always $\mathrm{E}\left(q^2\right) = \frac{S^2}{12}$. Finally, if we require a fixed power P of

the distortion signal $\mathbf{d} = \mathbf{x}' - \mathbf{x}$ after the DC post-processing, then S and α satisfy the following equation [EBTG03]:

$$P = \frac{\alpha^2 S^2}{12}. \tag{4.11}$$

The *capacity limit* of SCS under no attacks depends on the number of quantizers, i.e. the number of valid watermark symbols R, in the scheme. For example, the capacity limit of the binary SCS is 1 bit per (scalar) primitive x_i, while that of the 4-symbol SCS is 2 bits per primitive. From a practical viewpoint, the total number of the embedded watermark bits in SCS also depends on the number of watermarking primitives (i.e. the length of the host signal $\mathbf{x}$) and on the watermark channel coding method. For instance, if an M-bit watermark is embedded repetitively for 3 times in a $3M$-length host signal, then the watermarking capacity is counted as M bits instead of $3M$ bits. It is also interesting to study the capacity limit of SCS under attacks. In the case where the watermarked signal is corrupted by a source of additive Gaussian noise of power σ_n^2, Eggers et al. [EBTG03] showed that the capacity limit of SCS is quite close to that of the ideal Costa coding method as given in Equation (4.6) under weak to moderately strong noises (except for extremely weak noises for which the capacity is limited by the number of valid watermark symbols as discussed above). They also derived the optimum value of the DC factor that maximizes the watermarking capacity under Gaussian noise addition as follows:

$$\alpha_{SCS} = \sqrt{\frac{P}{P + 2.71\sigma_n^2}}, \tag{4.12}$$

where P and σ_n^2 are respectively the power of the distortion signal $\mathbf{d}$ and the noise $\mathbf{n}$. After obtaining α_{SCS}, we can then deduce the optimum quantization step size S_{SCS} by using the relationship given by Equation (4.11). Thus, it can be seen that when deriving the optimum SCS parameter values under attacks, we have only one degree of freedom (although it seems that there exist two adjustable parameters S and α) due to the premise of a fixed distortion. Finally, note that the optimum DC factor value α_{SCS} for SCS is different from that for the ideal Costa scheme (ICS) as provided by the following equation:

$$\alpha_{ICS} = \sqrt{\frac{P}{P + \sigma_n^2}}. \tag{4.13}$$

In general, the *robustness* of a quantization-based watermarking method depends mainly on the minimum distance between two codewords that represent different watermark symbols. Equivalently, by referring to Figure 4.2, we can see that the robustness intuitively relates to the sizes of the quantization cells. Therefore, the quantization step

S in SCS, which determines the size of the "quantization cells", is crucial to the watermark robustness. Meanwhile, under a fixed quantization step size, the robustness may be enhanced by increasing the DC factor α, especially under small-amplitude attacks. The reason is that the watermarked value $x_i^{'}$ is closer to the selected codeword that is located in the middle of the quantization interval, thus $x_i^{'}$ has a lower probability to be moved out of the decoding interval of the codeword under slight modifications.

The study of Eggers et al. [EBTG03] on the derivation of optimum α and S values under Gaussian noise addition which maximize the watermark bit transmission rate (presented above) provides some insights on the appropriate parameter setting of SCS in order to achieve better robustness under various attacks. However, in our opinion, the results of Eggers et al. seem not very practical in real-world applications for the following reasons. First, in most applications, we cannot precisely forecast the attacks that will occur, and meanwhile we are not allowed to adjust the watermark embedding parameters when the type or the amplitude of the endured attack changes. Normally, we have only one determinate stego-content for each purchaser, which is generated by using a set of determinate parameter values. Second, most of the existing work on optimum parameter setting of SCS is on the basis of a Gaussian noise attack. However, there exist many other attacks in real-world setting. For some attacks, it is very difficult to construct a mathematical model; consequently, the optimum value derivation for these attacks becomes very complicated or even impossible. Accordingly, it seems nearly infeasible to determine a set of values for α and S that are "optimum" for every attack. Hence, in practical applications, a common strategy is that we first fix the quantization step S for all the cover contents and all the scalar components of the host signal, and then we modify the DC factor α for each content and/or for each component, in order to achieve a satisfactory performance in terms of imperceptibility, robustness and security. The obtained robustness is satisfactory in a global sense, and obviously the watermark resistance to each kind of attack is not optimized.

The watermarking *security* [PFCTPPG06] is rather a high-level requirement. Recently, researchers tend to measure the security of a watermarking scheme by the amount of information leakage of the secret parameters of the scheme through observations (e.g. several watermarked copies generated by using the same set of secret parameters but with different watermark sequences embedded) [CFF05, CPFPG05]. In SCS watermarking, the secret parameters are the dither signal $t_{x_i}, i \in \{1, 2, ..., N\}$ which is generated by using a secret key K. Once having disclosed the dither signal, a pirate can perform a variety of malicious attacks such as unauthorized watermark extraction, optimal watermark

removal and watermark embedding in other contents by using this signal. Pérez-Freire et al. [PFCPG05] analyzed theoretically the security of the SCS watermarking scheme. For the binary SCS (we will mainly use this scheme in the following chapters), a perfect security, i.e. a perfect secrecy of the dither signal and thus the key, can be achieved if the DC factor is equal to 0.50, i.e. $\alpha = 0.50$, and then the security level decreases as α increases.

From the above presentation, we can see that when adjusting the binary SCS watermarking parameters R, S and α, it sometimes induces conflicting effects on the different watermark evaluation metrics. For example, if we increase the quantization step size S, then the robustness is generally improved but the induced distortion is also increased. Hence, in the design of SCS-based watermarking methods, it is important to find appropriate parameter values that lead to a satisfactory compromise between distortion, capacity, robustness and security.

4.4 Using SCS for Blind Mesh Watermarking

In order to use SCS for blind mesh watermarking, we should at least solve the following problems.

1. We should find an appropriate scalar watermarking primitive that will be subject to SCS quantization. The chosen primitive should have a very stable value under various attacks against which the watermarking method is required to be robust. Particularly, it seems not easy to construct a host signal space in which the mesh feature values are invariant to similarity transformations and robust against connectivity attacks. In addition, the quantization of the primitive value with a sufficient strength that ensures a good robustness should not introduce obvious distortion to the cover mesh.
2. We need to establish a robust synchronization (indexing) mechanism for the scalar components involved in the SCS quantization, and also avoid the causality problem from happening during the watermark embedding.
3. For a mesh watermarking scheme that embeds watermark in a non-invertible transformed space, we should derive an efficient method to reflect the scalar quantization in the host signal space on the vertex coordinate modifications in the spatial space.
4. It is difficult to find a set of universal SCS quantization parameters for mesh watermarking. Ideally, the quantization step size should be the same for different

mesh models and different scalar components, in order to facilitate the watermark extraction. In our SCS-based mesh watermarking methods, we adopt the following strategy to determine the values for the step size S and the DC factor α.

First, we try to find a fixed and appropriate S value for all the mesh models and all the scalar components, which roughly ensures a good trade-off between robustness and imperceptibility. However, in some cases, it is not possible to find such a universal quantization step size, mainly due to the strong diversity of the host signal space on different models and on different scalar components. Under this situation, we devise and adopt a variant of the conventional SCS to overcome this difficulty. One such example will be described in Chapter 6, in the design of the moment-based robust and blind mesh watermarking scheme. After this first step concerning the setting of the quantization step size, we can adjust the distortion compensation factor α for each model and sometimes also for each scalar component, in order to achieve a better performance. The above strategy is somewhat empirical but seems quite effective in practice.

In the presentation of our blind SCS-based watermarking methods in Chapters 5-7, we will elaborate on the specific strategy adopted by each method in order to solve the above problems.

Chapter 5

Hierarchical Watermarking of Semi-Regular Meshes Based on Wavelet Transform

Contents

This chapter presents a hierarchical and multiple watermarking framework for semi-regular meshes. In this framework, three blind watermarks are embedded in a same semi-regular mesh with different purposes: a geometrically robust watermark for copyright protection, a high-capacity watermark for carrying a large amount of auxiliary information, and a fragile watermark for content authentication. All the three watermarks are embedded in the wavelet domain of the semi-regular mesh through SCS quantization of the selected watermarking primitives. The chapter begins with an overview of the proposed hierarchical watermarking system. Then, the embedding and extraction algorithms of the three watermarks are presented. Finally, we provide some experimental results, analyze the algorithm performances and draw a conclusion.

The work described in this chapter was published in form of two international conference papers [WLDB07a, WLDB08b] and one international journal paper [WLDB08c].

5.1 Overview of the Hierarchical Watermarking Framework

As defined in Section 2.1, a 3-D mesh is called *regular* if all its vertices have a same valence. Consequently, a *semi-regular* mesh is a piecewise regular structure and consists of a patchwork of large regular regions; thus it owns regular vertices almost everywhere. A semi-regular mesh is often built starting from a coarse-level irregular mesh that is recursively refined through iterative subdivisions and displacements forming a multiresolution hierarchical structure. The inverse of this refining process is the wavelet decomposition of the obtained dense semi-regular mesh. The intrinsic multiresolution nature of the semi-regular meshes makes them very attractive in various applications involving level of details management, such as filtering, texturing, rendering and particularly compression where a lot of work has been done [KSS00, PA06] even for dynamic mesh sequences [YKL06]. Accordingly, even very recently a lot of remeshing techniques have been proposed for constructing such multiresolution semi-regular models starting from 3-D volumetric models [WSBD00] or irregular meshes [Gus07] even gigantic [AGL06]. Along with the more and more popular use of these semi-regular meshes, their intellectual property protection and authentication problems have attracted more and more attention. Naturally, as promising techniques, robust and fragile watermarking algorithms appear as good candidates to solve these problems. Meanwhile, a high-capacity watermark is sometimes very useful to carry a large amount of auxiliary information, such as the mesh generation information, a description, a related website address, or even animation parameters.

In our hierarchical watermarking framework, three different watermarks (robust, high-capacity and fragile) are embedded in a same semi-regular mesh, serving for different applications (copyright protection, content enhancement and content authentication). These applications are not mutually exclusive. For instance, we can imagine the following scenario: a manufacturer designs a complex car part represented by a semi-regular mesh, then he may wish to embed in this part a piece of copyright information for intellectual property protection against possible forgery; he may also want to insert a fragile watermark so as to ensure that any illegal modification can be easily detected by authorized clients; and finally he may like to embed into the object some description information, such as the part design norm and its applicable car models. Indeed, the concept of multiple (or multipurpose) watermarking [MB99, SSNO01] has been investigated for a long time and several techniques have been proposed for images [LL01] and audio clips [LLC00]. To the best of our knowledge, this chapter presents the first attempt on multiple watermarking for 3-D meshes. According to the paper of Sheppard

et al. [SSNO01], in general, a multiple watermarking system seems as secure as the individual underlying algorithms; on the contrary, the robustness, imperceptibility and capacity of individual underlying algorithms are generally degraded by the embedding of the other watermarks. In the proposed hierarchical and multiple watermarking system, there is not any interference between the different embedded watermarks so that their individual performances are kept as much as possible.

The watermark embedding space in our system is the wavelet domain of the host semi-regular mesh [LDW97] (c.f. Section 3.2.1.2 that is entitled by "Fragile techniques in transform domain"). Here, we briefly recall the principle of the wavelet analysis of semi-regular meshes: when passing from a dense level to a coarse level, the *even* vertices are conserved while the *odd* vertices are removed (c.f. Figure 3.3); the *wavelet coefficient vectors* (WCVs) are calculated as the prediction errors for all the removed odd vertices and they are 3-D vectors associated with each edge of the coarse-level mesh. Wavelet analysis can be iteratively applied on a dense mesh with semi-regular subdivision connectivity, and the dual wavelet synthesis algorithm can accomplish the inverse reconstruction. Indeed, the mesh multiresolution analysis based on wavelet transform is a very suitable tool for constructing a hierarchical multiple watermarking system: first, there is no interference between different watermarks if they are embedded in the WCVs of different levels; secondly, and also more importantly, these watermarks can be embedded at different appropriate resolution levels according to their specific objectives.

Figure 5.1 illustrates the proposed hierarchical watermarking framework: the fragile watermark is embedded in a dense resolution level obtained just after one wavelet decomposition of the original mesh, by modifying the orientations and norms of the corresponding WCVs; the robust watermark is inserted by modifying the norms of the WCVs associated with the lowest resolution level; the high-capacity watermark is embedded in one or several intermediate levels by considering groups of WCV norms as watermarking primitives. In practice, the robust watermark is first embedded after a thorough decomposition, then the high-capacity watermark and the fragile watermark are embedded successively during the reconstruction procedure. This workflow effectively prevents the posteriorly embedded watermark from impacting the anteriorly embedded one(s) and follows the principle proposed in [MB99], which points out that the most robust watermark should be embedded in the first place while the most fragile one should be embedded at the last. By using this embedding order, we also make assumption that the insertion of the first two watermarks does not obviously degrade the functional quality (especially for CAD objects [OM01]) and the perceptual quality of the mesh, so as to make the authentication based on the fragile watermark meaningful.

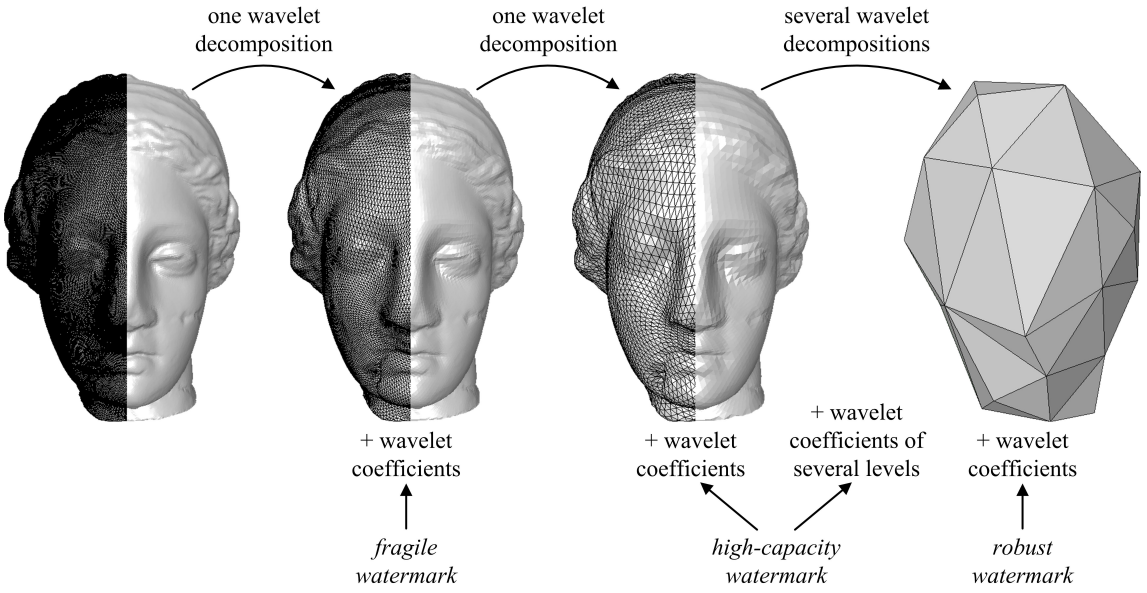


Figure 5.1: Illustration of the proposed hierarchical multiple watermarking framework.

All the three watermarks in our system are blind and invariant to the so-called content-preserving attacks including vertex/facet reordering and similarity transformation, which theoretically do not have any influence on the mesh shape. The robust watermark is able to resist all the common geometry attacks even with relatively strong amplitude. The fragile watermark is robust to the aforementioned content-preserving attacks. However, it is vulnerable to others attacks such as local and global geometry modifications since the objective is to check the integrity of the semi-regular mesh. In addition, at extraction, these attacks can be precisely located on the surface of the attacked mesh in a blind way. The payload of the high-capacity watermark increases rapidly with the increase of the number of watermarking primitives.

5.2 Blind and Robust Watermark

5.2.1 Objective and basic idea

In our opinion, a robust watermark for semi-regular meshes may not have to be resistant to the connectivity attacks, since these attacks normally destroy the semi-regular multiresolution connectivity thus the intrinsic attractiveness of such meshes. Therefore, the objective in this section is to construct a blind watermark that is robust against all the common geometry attacks.

In general, the robustness of a blind mesh watermarking scheme mainly depends on two aspects (c.f. Section 3.1): 1) the robustness of the watermarking primitive feature

values in which the watermark is embedded and 2) the robustness of the watermark synchronization mechanism. Concerning the first point, we choose to embed the robust watermark by modifying the norms of the WCVs associated with the coarsest-level mesh obtained after a thorough wavelet decomposition of the original dense semi-regular mesh. These WCV norms are of relatively low frequency and are supposed to be robust against geometry attacks. Concerning the second point, our proposal is to use a certain robust aspect to synchronize the embedded watermark bits: the edges in the coarsest-level mesh are sorted according to their lengths; this order is experimentally very robust against geometry attacks. The watermark bits are successively embedded through scalar Costa quantization (c.f. Chapter 4) of the norms of the WCVs associated with these sorted edges (c.f. Figure 3.3). Moreover, in this way, the synchronization primitives (edge lengths) and the watermarking primitives (WCV norms) are separated, so the causality problem is avoided.

5.2.2 Watermark embedding

The first step of the embedding procedure is to carry out a thorough wavelet decomposition (supposing that it consists of J iterations) on the original non-watermarked semi-regular mesh $\mathcal{M}_0$. Then, we obtain a coarsest-level irregular mesh $\mathcal{M}_J$ and J sets of WCVs. For robust watermark embedding, we only consider the set of N_J WCVs associated with $\mathcal{M}_J$, where N_J is also the number of edges in $\mathcal{M}_J$.

In the next step, all the edges in $\mathcal{M}_J$ are sorted according to the descending order of their lengths. Thus, the longest edge in $\mathcal{M}_J$ is denoted by $\mathbf{e}_1^J$, the second longest edge is denoted by $\mathbf{e}_2^J$, and so forth. The WCV associated with $\mathbf{e}_1^J$ is denoted by $\mathbf{c}_1^J$, the one associated with $\mathbf{e}_2^J$ is denoted by $\mathbf{c}_2^J$, etc. Thus, an order of all the edges (also of all the WCVs) has been established. Note that $\mathbf{e}_i^J, i \in \{1, 2, ..., N_J\}$ is defined as the coordinate difference of its two incident vertices and thus is considered as a 3-D vector.

The watermark is an M-length bit sequence $w_1, w_2, ..., w_M$ with $w_i \in \{0, 1\}, i \in \{1, 2, ..., M\}$. These bits are embedded through SCS quantization of the norms of $\mathbf{c}_i^J, i \in \{1, 2, ..., N_J\}$. First of all, we have to fix a quantization step S_{rob} for these norms: the average edge length of this resolution level is calculated as $\bar{l}^J = \frac{1}{N_J} \sum_{i=1}^{N_J} \left\| \mathbf{e}_i^J \right\|$ and S_{rob} is fixed as $\bar{l}^J / \epsilon_{rob}$, where ϵ_{rob} is a control parameter to achieve an expected trade-off between robustness and imperceptibility. An appropriate value of ϵ_{rob} can be found through experimental study so that it can be fixed for most of the semi-regular meshes without seriously affecting the algorithm's performances.

The next step is the watermark bit embedding. The bit w_i is embedded by quantizing

$\left\|\mathbf{c}_i^J\right\|$ through the 2-symbol scalar Costa scheme, i.e. the binary SCS (c.f. Chapter 4). The practical quantization procedure is as follows: first, a component-wise random codebook is established for each $\left\|\mathbf{c}_i^J\right\|$ as given by Equation (5.1), where S_{rob} is the pre-fixed quantization step, $z \in \mathbb{Z}$ is an integer, $l \in \{0,1\}$ is a watermark symbol (here a bit), and $t_{\|\mathbf{c}_i^J\|}$ is the i-th component of an N_J-length additive pseudo-random dither signal.

$$\mathcal{U}_{\|\mathbf{c}_i^J\|,t_{\|\mathbf{c}_i^J\|}} = \bigcup_{l=0}^{1}\left\{u = zS_{rob} + l\frac{S_{rob}}{2} + t_{\|\mathbf{c}_i^J\|}S_{rob}, u \geq 0\right\}. \tag{5.1}$$

Each codeword u in $\mathcal{U}_{\|\mathbf{c}_i^J\|,t_{\|\mathbf{c}_i^J\|}}$ represents a watermark bit, which is the value of l in u's derivation. The N_J-length dither signal is generated by using a secret key K_{rob} and is introduced to achieve randomization of the codebooks for the WCV norms. In our experiments, $t_{\|\mathbf{c}_i^J\|}, i \in \{1,2,...,N_J\}$ form a simulation sequence of a random variable T_{rob} that follows the uniform distribution between $\left(-\frac{1}{2},\frac{1}{2}\right)$ (i.e. $T_{rob} \sim U\left(-\frac{1}{2},\frac{1}{2}\right)$), and they can be generated by inputting K_{rob} to an appropriate pseudo-random number generator.

Then, we find the nearest codeword $u_{\|\mathbf{c}_i^J\|}$ to $\left\|\mathbf{c}_i^J\right\|$ in this codebook that correctly represents the to-be-embedded watermark bit w_i (i.e. w_i should be equal to the l's value in the expression of $u_{\|\mathbf{c}_i^J\|}$ as given in the braces in Equation (5.1)). The quantized value $\left\|\mathbf{c}_i^J\right\|'$ is calculated according to Equation (5.2), where $\alpha_{rob} \in [0,1]$ is the distortion compensation (DC) factor. Usually, we choose $\alpha_{rob} \geq 0.50$ in order to ensure the correctness of the watermark extraction under no attacks.

$$\left\|\mathbf{c}_i^J\right\|' = \left\|\mathbf{c}_i^J\right\| + \alpha_{rob}\left(u_{\|\mathbf{c}_i^J\|} - \left\|\mathbf{c}_i^J\right\|\right). \tag{5.2}$$

The induced distortion and the security of the robust watermark are in part driven by the value of α_{rob}. Recall that for the binary SCS, a perfect security can be achieved if the DC factor is equal to 0.50 and then the security level decreases as α increases [PFCPG05].

Finally, keeping its orientation unchanged, we modify the length of $\mathbf{c}_i^J$ to realize the norm quantization. If the edge number N_J is greater than the watermark bit number M, a redundant embedding will be carried out in order to enhance the watermarking robustness. Two repetition schemes are possible: the first is to sequentially divide the ordered edges in several groups each having M edges, and the watermark sequence is repeatedly embedded in each group; the second is to sequentially divide the edges in M equal parts and repeatedly embed one bit in each part. The second scheme is experimentally less robust due to the vulnerability of the last few watermark bits that are embedded in the shortest edges. The insertion in these short edges is naturally less robust than that in the longer ones since their associated WCVs are usually of higher

frequencies. Hence, the first repetition scheme was adopted.

Once the quantization of all the WCVs is accomplished, we apply wavelet synthesis on $\mathcal{M}_J$ with the modified WCVs until the resolution level where the high-capacity watermark is to be embedded.

Algorithm 5.1 summarizes the blind and robust watermark embedding procedure.

1 Do wavelet analysis of the original semi-regular mesh until the coarsest level
2 Do descending sort of all the edges in this level according to their lengths
3 Calculate the average length $\bar{l}^J$ of the edges and fix the WCV norm quantization step as $\bar{l}^J / \epsilon_{rob}$
4 **for** each edge in the descending sort **do**
5 Calculate the norm of its associated WCV
6 Quantize this norm according to Equation (5.2) by using the 2-symbol scalar Costa quantization scheme
7 **end for**
8 Do mesh reconstruction until the level where the high-capacity watermark is to be embedded

Algorithm 5.1: Blind and robust watermark embedding procedure.

5.2.3 Watermark extraction

With the knowledge of the secret key K_{rob} used during the watermark embedding, the watermark extraction is blind and quite simple. It is sufficient to carry out a thorough wavelet analysis, reestablish the edge order, calculate the quantization step, reconstruct the component-wise codebook for each WCV and finally find out its represented bit by looking for the nearest codeword in this codebook to the actual value of the WCV norm. If redundant embedding is used during watermark embedding, a simple majority voting strategy is adopted at extraction to deduce the watermark bit values.

5.2.4 Analysis and discussion

The robust watermark presented above is theoretically invariant to similarity transformations, because the WCV norm quantization step S_{rob} is proportional to $\bar{l}^J$, the average length of all the edges in the coarsest-level mesh $\mathcal{M}_J$. Thus, the real watermarking primitive can be equivalently considered to be the ratio between the norm of a WCV and the average edge length $\bar{l}^J$, which is a similarity-transformation-invariant quantity.

The watermark induced distortion for each odd vertex (c.f. Figure 3.3) in the reconstructed $(J-1)$-level mesh $\mathcal{M}'_{J-1}$ is the norm difference between the quantized and

the original WCVs that represent the prediction error of the odd vertex. It is easy to deduce that the upper limit of this distortion is equal to $\alpha_{rob}.\frac{S_{rob}}{2}$. This distortion will later propagate to the odd vertices introduced by the following reconstruction steps.

The robust watermarking scheme fails if the synchronization mechanism fails or if the watermark bit SCS decoding fails. The former usually demonstrates stronger robustness than the latter under moderate and strong geometry attacks. Obviously, the watermark will generally be destructed under connectivity attacks, which yet can be omitted in semi-regular mesh watermarking, as mentioned at the beginning of this section. If we want also the robustness against connectivity attacks, one possible solution is to devise a robust remeshing technique that is insensitive to connectivity changes. Before watermark extraction, the attacked mesh is first remeshed to reconstruct a semi-regular mesh with the same connectivity configuration as the one in which the watermark is initially embedded. Such a remeshing technique could possibly rely on a blind and robust feature points detection algorithm but its development seems difficult. Rondao-Alface et al. [RAM05] have conducted some related work on this research problem. One special connectivity attack is the cropping. We guess that a partial wavelet analysis is still possible on the intact regions of a cropped semi-regular mesh, so that the watermark can still be successfully extracted because of the redundant embedding. The principle would be first calculating the autocorrelation function of the extracted bit sequence from the intact parts of the coarsest-level mesh, then it would be possible to resynchronize and retrieve the watermark according to the cyclic peaks of this autocorrelation function.

One limitation of the proposed robust scheme is that it will probably fail for the regular or semi-regular meshes where the edges in the coarsest-level representation have almost the same length. Under this situation, other metrics have to be used to sort these coarsest-level edges, such as the areas or the roughness of the regions [Lav09] in the original dense mesh that correspond to the incident facets of the coarsest-level edges.

5.3 Blind and High-Capacity Watermark

In this section, we describe a new high-capacity watermarking scheme for semi-regular meshes. The watermark is embedded through permutation alteration of the selected geometric primitive and is invariant to all the content-preserving operations.

5.3.1 Watermark embedding

For a mesh $\mathcal{M}_H^{'}$ at a certain level of the wavelet synthesis procedure carried out after the robust watermark embedding, we suppose that its N_H WCVs are indexed according

Table 5.1: Example of the high-capacity watermark embedding steps ($N_H = 5$).

Edges lengths	3.2	3.0	2.7	2.1	1.8
Edge / WCV indices (i for $\mathbf{c}_i^H$ and $\mathbf{e}_i^H$)	1	2	3	4	5
WCV norms ($\left\|\mathbf{c}_i^H\right\|$)	0.28	0.35	0.24	0.21	0.22
Residues of the norms divided by $p = 0.1$ ($res\,(i)$)	0.08	0.05	0.04	0.01	0.02
Original WCV orders ($order_o\,(i)$)	5	4	3	1	2
Expected WCV orders ($order\,(i)$)	3	4	5	2	1
New residues ($\frac{order(i).p}{N_H+1}$)	0.0500	0.0667	0.0833	0.0333	0.0167
New WCV norms ($\left\|\mathbf{c}_i^H\right\|'$)	0.2500	0.3667	0.2833	0.2333	0.2167

to the lengths of their associated edges in $\mathcal{M}_H^{'}$, in the same way as in the last section. This means that the WCV indexed by i is associated with the i-th longest edge in $\mathcal{M}_H^{'}$.

Then we combine each WCV $\mathbf{c}_i^H$ with another number denoted by $order_o\,(i)$. To obtain this number, we first calculate the residue of the norm $\left\|\mathbf{c}_i^H\right\|$ divided by a control parameter p as $res\,(i) = \left\|\mathbf{c}_i^H\right\| \%p$; $order_o\,(i)$ is the ascending order of the value $res\,(i)$ among the residues of all the WCVs at the same level. Similar to the quantization step size S_{rob} used in the robust watermarking scheme, the control parameter P is fixed as $\bar{l}^H/\epsilon_{hc}$ and is also related to the average length of the edges (but at a different resolution level H). The first five lines of Table 5.1 provide one simple example of this calculation, where $N_H = 5$ and $p = 0.1$. For instance, $res\,(1)$ of $\mathbf{c}_1^H$ is equal to 0.08, which is the largest among all the residues of the five WCVs, thus $order_o\,(1)$ is set to be 5.

These order numbers are listed successively as $order_o\,(1)$,$order_o\,(2)$,...,$order_o\,(N_H - 1)$, $order_o\,(N_H)$, along with the ascending order of the index i (as shown by the fifth line of Table 5.1). This sequence is a permutation of the N_H numbers ranging from 1 to N_H and thus has $N_H!$ different possibilities. As a consequence, each permutation can potentially represent a watermark of $\lfloor \log_2 (N_H!) \rfloor$ bits (i.e. the largest integer less than or equal to $\log_2 (N_H!)$). The correspondence between the watermarks ($\lfloor \log_2 (N_H!) \rfloor$-length bit strings) and the possible order sequences (N_H-number permutations) is established according to the following rule: for two permutations, the one with a bigger first number (from left) represents a bigger bit string (in terms of its binary value); and if the first numbers are the same, we compare the second, and so on. Under this rule, the permutation $1, 2, 3, ..., N_H - 1, N_H$ represents the smallest bit string $0, 0, ..., 0, 0$; and the permutation $1, 2, 3, ..., N_H, N_H - 1$ represents the second smallest bit string $0, 0, ..., 0, 1$.

Under this rule, each possible watermark bit string can be represented by a permutation. Thus, it seems natural to substitute the original permutation by a new one in order to embed a given watermark. This new permutation is established by modifying the WCV norms so as to alternate their norm residues' orders. The new WCV norm is

determined by Equation (5.3), where $order\,(i)$ is the new expected norm residue order of the WCV $\mathbf{c}_i^H$ that is associated with the i-th longest edge.

$$\left\|\mathbf{c}_i^H\right\|' = \left\lfloor \frac{\left\|\mathbf{c}_i^H\right\|}{p} \right\rfloor .p + \frac{order\,(i)\,.p}{N_H + 1}. \tag{5.3}$$

The last three lines of Table 5.1 provide one simple example of this substitutive watermark embedding procedure. It can be observed that only the residue of the WCV norm is substituted, while the difference between the WCV norm and the residue is kept unchanged.

Practically, the N_H edges are divided into several ordered groups of G edges (in each group are embedded $\lfloor \log_2 (G!) \rfloor$ bits), in order to make the watermark less fragile. Thus, the practical capacity of this method is $\left\lfloor \frac{N_H}{G} \right\rfloor . \lfloor \log_2 (G!) \rfloor$ bits. The simplest grouping is adopted: putting the edges indexed by 1 to G in the first group, the ones indexed by $(G+1)$ to $2G$ in the second group, and so forth. Algorithm 5.2 summarizes the embedding procedure of the proposed high-capacity watermarking scheme. Note that in step 8, it would be possible to deduce the new WCV norms by using the scalar Costa scheme, with the distortion compensation and the introduction of a dither signal that is generated by a secret key K_{hc}. In this way, the induced distortion is decreased and the watermarking security is enhanced.

1 Do wavelet synthesis after robust watermark embedding until a certain appropriate level H
2 Do descending sort of all the edges at this level according to their lengths
3 Calculate the average length $\bar{l}^H$ of these edges and determine the control parameter p as $\bar{l}^H / \epsilon_{hc}$
4 Divide the edges in several ordered groups of G edges according to their length sorting
5 **for** each ordered edge group **do**
6 Translate the next $\lfloor \log_2 (G!) \rfloor$ bits in the watermark sequence to a corresponding permutation
7 **for** each descending sorted edge in the current group **do**
8 Substitute the norm of its associated WCV according to Equation (5.3) so as to assign to it an expected norm residue order in the desired permutation
9 **end for**
10 **end for**
11 Do mesh reconstruction until the second densest level where the fragile watermark is to be embedded

Algorithm 5.2: Blind and high-capacity watermark embedding procedure.

5.3.2 Watermark extraction

Like the robust watermark, the extraction of the high-capacity watermark is simple and blind. After dividing the edges in several ordered groups and for each group establishing a permutation according to the WCV norm residues' sorting, we can find out the watermark bit substring represented by each group. Finally, all the extracted substrings are concatenated to form the complete watermark bit string.

5.3.3 Analysis and discussion

If the WCVs in all the J resolution levels are used for high-capacity watermarking (with no robust and fragile watermarks embedded), the capacity upper limit of our method is $\left\lfloor \frac{N_0^v - N_J^v}{G} \right\rfloor . \lfloor \log_2 (G!) \rfloor$, where N_0^v and N_J^v are the numbers of vertices in $\mathcal{M}_0$ and $\mathcal{M}_J$, respectively (remember that each removed odd vertex has a corresponding WCV). Considering that N_J^v is normally negligible compared to N_0^v, the capacity limit can thus be approximated by $\left\lfloor \frac{N_0^v}{G} \right\rfloor . \lfloor \log_2 (G!) \rfloor$. If we want a higher capacity, the other two degrees of freedom (DF) of the WCV (e.g. the two angles of the WCV in the local spherical coordinate system) can also be watermarked independently from its norm by using a similar permutation-based scheme.

As mentioned in Section 3.2.2, the existing high-capacity mesh watermarking schemes can be classified as geometry-based or order-based methods. The latter normally has a much higher capacity but it is fragile to vertex/facet reordering. Meanwhile, the former can easily achieve the robustness against all the content-preserving operations but its capacity is somewhat limited. The proposed high-capacity watermarking scheme combines the ideas of both kinds of methods. Indeed, our watermark is embedded in a geometric primitive by following the basic idea of the permutation-based steganography technique [Arto1]. Compared to the existing geometry-based methods, our scheme achieves a higher capacity (c.f. the quantitative comparison in the next paragraph). Ideally, the proposed watermark can achieve the optimum capacity of the permutation-based data hiding methods (i.e. $\log_2 n!$ bits on an n-element set), if the vertex coordinates have a sufficient (and extremely high) precision. However, the practical capacity is of course lower than this optimum value due to the calculation and storage precision limits of the vertex coordinates. The advantage of our scheme over the existing order-based methods is its invariance to vertex/facet reordering.

Figure 5.2 graphically compares the capacity of our method with those of some existing geometry-based high-capacity methods (without any robustness consideration): Cayre and Macq's method (1 bit/vertex) [CM03], Benedens' method (1 bit/facet) [Ben99b],

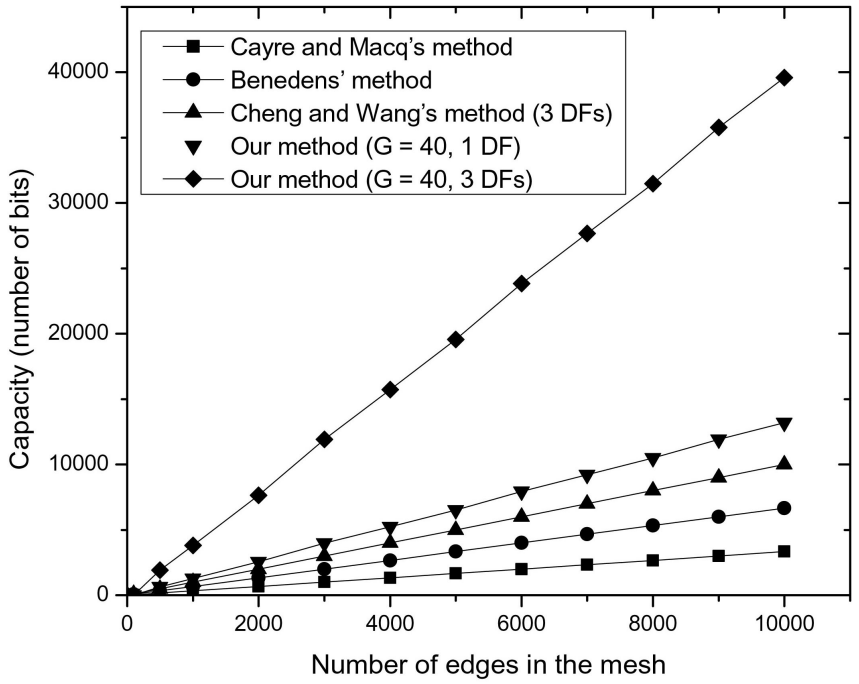


Figure 5.2: Capacity comparison of different high-capacity methods.

Cheng and Wang's method (3 bits/vertex, 3 DFs) [CW06], and our methods ($G = 40$ with only the WCV norm watermarked and with all the three WCV DFs watermarked). For this comparison, it is assumed that for a manifold triangular mesh, we usually have $N_E = 1.5N_F$ and $N_F \approx 2N_V$, where N_V, N_E, N_F are respectively the numbers of vertices, edges, and facets in the dense mesh (c.f. Section 2.1).

If the control parameter p here is equal to the quantization step size S_{rob} in the robust watermarking algorithm that is described in the last section, then the WCV norm distortions in the worst situation are of the same magnitude in these two algorithms. However, the value p is subdivided into G subintervals instead of 2 for S_{rob}, and meanwhile the high-capacity watermark relies on the relationship between the norms of different WCVs. Therefore, even a slight norm attack much smaller than p would seriously disturb the established WCV norm orders. That is the principal reason for the relative fragility of the proposed high-capacity watermarking scheme.

5.4 Fragile Watermark

We recall that the objective of the fragile watermark is to be invariant to all the content-preserving operations while being vulnerable to the other attacks. Meanwhile, the endured attacks have to be precisely located on the surface of the attacked mesh according to the watermark extraction result. Finally, the blindness of the watermark extraction,

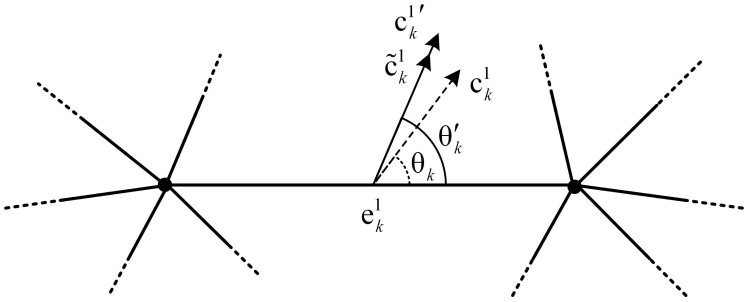


Figure 5.3: Illustration of the fragile watermarking primitives and the modification of the orientation and the norm of a WCV.

which is mandatory for an authentication algorithm, has also to be achieved.

5.4.1 Watermark embedding

The first step is to carry out the wavelet synthesis after robust and high-capacity watermark embeddings until the second densest level (i.e. the level 1). Then, we obtain a relatively dense mesh $\mathcal{M}_1^{'}$ and a set of N_1 WCVs denoted by $\mathbf{c}_1^1, \mathbf{c}_2^1, ..., \mathbf{c}_{N_1}^1$. Each WCV $\mathbf{c}_k^1, k \in \{1, 2, ..., N_1\}$ is associated with an edge $\mathbf{e}_k^1$ in $\mathcal{M}_1^{'}$. Note that, differently from in the last two sections, the fragile watermark embedding procedure is independent from these indices so they can be assigned arbitrarily. In our algorithm, we take the edges in the less dense mesh $\mathcal{M}_1^{'}$ as raw authentication primitives; then we derive the validity of each vertex in the watermarked (and probably attacked) dense mesh $\mathcal{M}_0^{'}$ based on the authentication results of these edges.

The basic idea of the watermark embedding is to find two watermarking primitives for each edge $\mathbf{e}_k^1$ and then slightly modify them in order to embed in both of them a same watermark symbol s_k. Thus, each edge is made valid for authentication by establishing an equality relationship between the two symbols represented by the two modified primitives. Ideally, these two primitives have to be modified independently, and the primitives of different edges have also to be modified independently. In this way, the causality problem (within an individual edge and between different edges) is prevented and the invariance to vertex/facet reordering is attained. We have found two such primitives: the first one is the acute angle between $\mathbf{c}_k^1$ and $\mathbf{e}_k^1$ that is denoted by θ_k as illustrated in Figure 5.3; the second one is the ratio between the norm of $\mathbf{c}_k^1$ and the length of $\mathbf{e}_k^1$ that is denoted by $r_k = \left\|\mathbf{c}_k^1\right\| / \left\|\mathbf{e}_k^1\right\|$. Both primitives are invariant to similarity transformations.

The next step is the watermark symbol embedding. This symbol s_k can be any of the item in the symbol set (alphabet) $\mathcal{W} = \{w_1, w_2, ..., w_L\}$, where L is the number of

legal symbols. θ_k and r_k are both quantized by using the L-symbol scalar Costa scheme. θ_k is first quantized; as shown in the following, its quantization does not modify the symbol represented by its initial value. Indeed, the objective here is to find this initially represented symbol and fix it as s_k for the edge $\mathbf{e}_k^1$ and therefore for the future quantization of r_k. This quantization also ensures a sufficient robustness of the represented symbol of θ_k to similarity transformations, which can cause slight perturbation of θ_k due to calculation and storage precision limits. The reason for performing symbol-preserving quantization on θ_k rather than on r_k is that θ_k is more sensitive to similarity transformations and that its modification is less imperceptible than r_k.

The quantization of θ_k is realized by using the L-symbol scalar Costa scheme (c.f. Chapter 4) and its procedure is described as follows: first, a component-wise pseudo-random codebook is established for each θ_k as given by Equation (5.4), where S_θ is the quantization step, $z \in \mathbb{Z}$ is an integer, $l \in \mathcal{L} = \{0, 1, ..., L-1\}$ each stands for one of the L legal symbols in $\mathcal{W}$ (the bijective mapping between $\mathcal{L}$ and $\mathcal{W}$ is determined by a secret key K_{m_1}; this mapping is introduced to prevent evident watermark forgery while trying to modify the watermarked mesh), and t_{θ_k} is a pseudo-random dither value. Note that each codeword u in $\mathcal{U}_{\theta_k, t_{\theta_k}}$ represents a symbol in $\mathcal{W}$ that is the mapped symbol of the value l in u's derivation.

$$\mathcal{U}_{\theta_k, t_{\theta_k}} = \bigcup_{l=0}^{L-1} \left\{ u = zS_\theta + l\frac{S_\theta}{L} + t_{\theta_k} S_\theta, 0^\circ \leq u \leq 90^\circ \right\}. \tag{5.4}$$

Then we find the nearest codeword u_{θ_k} to θ_k in this codebook and take its represented symbol as s_k. The quantized value $\theta_k^{'}$ is calculated according to Equation (5.5), where $\alpha_\theta \in [0, 1]$ is a DC factor.

$$\theta_k^{'} = \theta_k + \alpha_\theta \left(u_{\theta_k} - \theta_k \right). \tag{5.5}$$

Finally, as shown by Figure 5.3, the orientation of $\mathbf{c}_k^1$ is modified by rotating it around the midpoint of $\mathbf{e}_k^1$ in the 2-D plane engendered by $\mathbf{c}_k^1$ and $\mathbf{e}_k^1$, to obtain an intermediate temporary vector $\tilde{\mathbf{c}_k^1}$ that reaches the expected angle value $\theta_k^{'}$.

Like $t_{\|\mathbf{c}_i^j\|}$ in Equation (5.1), t_{θ_k} is also introduced to achieve randomization of the codebook. Usually, an ordering for all the watermarking primitives is established and the generated pseudo-random numbers can then be assigned one by one to the ordered primitives, such as in the last two sections concerning the robust and the high-capacity watermarks. However, we cannot adopt such a mechanism for θ_k, because we want a precise attack localization capability of the fragile watermark, for which a global ordering (synchronization) of the watermarking primitives is not appropriate. To resolve this issue, we consider a local geometric ratio gr_k, between $\mathbf{e}_k^1$'s length and the length sum

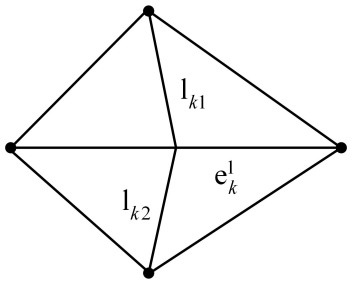


Figure 5.4: The geometric ratio gr_k used to construct the look-up tables. In a manifold mesh, each edge $\mathbf{e}_k^1$ is incident to two facets (one for a border edge), and $\mathbf{l}_{k1}$, $\mathbf{l}_{k2}$ are the midlines of these facets passing the midpoint of $\mathbf{e}_k^1$. gr_k is thus calculated as the following length ratio: $gr_k = \left\|\mathbf{e}_k^1\right\| / \left(\|\mathbf{l}_{k1}\| + \|\mathbf{l}_{k2}\|\right)$.

of $\mathbf{e}_k^1$'s incident triangles' midlines that pass the midpoint of $\mathbf{e}_k^1$ (c.f. Figure 5.4). Note that this geometric ratio is invariant to similarity transformations. A look-up table is introduced, which gives the correspondence between the value ranges of this geometric ratio and the sequential pseudo-random numbers generated by using a secret key K_θ. In our implementation, these pseudo-random numbers form a simulation sequence of a uniformly-distributed random variable $T_\theta \sim U\left(-\frac{S_\theta}{2L}, \frac{S_\theta}{2L}\right)$. The gr_k value ranges in the look-up table can further be scrambled by another key K_{t_θ} to enhance the security. For each θ_k, a number is selected from this table as t_{θ_k} according to the real value of gr_k.

The quantization of the norm-length ratio r_k is quite similar, with the usage of an appropriate quantization step size S_r and three secret keys K_{m_2}, K_r and K_{t_r}. The significant difference is the use of a constrained codebook (given by Equation (5.6) where l_{s_k}'s mapped symbol is s_k) to carry out the quantization so that the quantized value $r_k^{'}$ represents the same symbol s_k as $\theta_k^{'}$.

$$\mathcal{U}_{s_k,r_k,t_{r_k}} = \left\{ u = zS_r + l_{s_k}\frac{S_r}{L} + t_{r_k}S_r, u \geq 0 \right\}. \tag{5.6}$$

Keeping the orientation of $\tilde{\mathbf{c}}_k^1$ unchanged, we can modify its norm in order to obtain the watermarked WCV $\mathbf{c}_k^{1^{'}}$ that also reaches the expected norm-length ratio value $r_k^{'}$. Note that all the terms involved in the quantizations (i.e. θ_k, r_k and gr_k) are local to edge $\mathbf{e}_k^1$ and independent from any element ordering; hence, the precise attack localization capability and the invariance to element reordering are ensured.

Once the two quantization procedures are accomplished, an equality relationship between the two embedded watermark symbols has been established for each edge $\mathbf{e}_k^1$ in $\mathcal{M}_1^{'}$. Then a watermarked dense mesh $\mathcal{M}_0^{'}$ can be reconstructed by applying one wavelet synthesis on $\mathcal{M}_1^{'}$ with the modified WCVs $\mathbf{c}_k^{1^{'}}, k \in \{1, 2, ..., N_1\}$.

To summarize, Algorithm 5.3 lists the main steps of the fragile watermark embed-

ding procedure.

1 Do wavelet synthesis after robust and high-capacity watermark embeddings until level 1
2 Generate two pseudo-random dither signals t_{θ_k} and t_{r_k} by using two secret keys K_θ and K_r
3 Construct two look-up tables providing correspondences between the value ranges of the geometric ratio gr_k and the pseudo-random dither signals t_{θ_k} and t_{r_k}
4 **for** each edge $\mathbf{e}_k^1$ in this resolution level **do**
5 Do symbol-preserving SCS quantization for θ_k with the established look-up table between gr_k and t_{θ_k}
6 Do SCS quantization for r_k with the look-up table between gr_k and t_{r_k}, so that the two quantized values $\theta_k^{'}$ and $r_k^{'}$ both represent a same watermark symbol s_k
7 **end for**
8 Do one iteration of wavelet synthesis to obtain the watermarked dense mesh $\mathcal{M}_0^{'}$

Algorithm 5.3: Fragile watermark embedding procedure.

5.4.2 Watermark extraction and mesh authentication

The first step is to carry out one wavelet decomposition of the semi-regular mesh to be authenticated. For each edge $\mathbf{e}_k^1$ in the obtained less dense mesh, we construct two codebooks (respectively for θ_k and r_k) by using the acquired secret keys. Two watermark symbols can then be easily extracted by seeking the nearest codewords in the constructed codebooks to the actual values of θ_k and r_k. If these two symbols are equal, the current edge $\mathbf{e}_k^1$ is marked as valid; otherwise as invalid.

Then the task is to derive the validity for each vertex in the dense mesh. The validity for an even vertex in the dense mesh (c.f. Figure 3.3) is determined at first by using the following rule: if any of its incident edges in the less dense mesh is invalid, then it is considered as invalid; otherwise as valid. The validity of an odd vertex in the dense mesh (c.f. Figure 3.3) is then determined according to the validities of its two neighboring even vertices: if either of these two vertices is invalid, it is considered as invalid; otherwise as valid. We adopt such a mechanism in order to handle the *false positive* issue under attacks. Actually, each edge in the less dense mesh has a false positive probability (i.e. the edge is considered as valid but in fact it is not) that is about $\frac{1}{L}$ under attacks. By using the above decision rule, the false positive rate for an even vertex (suppose that it is of valence 6) is decreased to about $\left(\frac{1}{L}\right)^6$ if it is in the middle of an attacked region, and that of an odd vertex is also considerably decreased (to about $\left(\frac{1}{L}\right)^{11}$ if it is in the middle of an attacked region). Under attacks, it is also

possible to encounter the *false negative* issue, which means that some valid vertices may be mistakenly marked as invalid. As in the existing fragile mesh watermarking schemes [LLLL05, CT06], in our method the false negative issue only concerns the vertices that are neighboring to the real invalid vertices and seems inevitable if we want the invariance to similarity transformations.

Finally, the authentication results of the vertices in the dense semi-regular mesh are visualized to the users.

5.4.3 Analysis and discussion

The upper limit of the distortion induced by the fragile watermark embedding on the odd vertices in $\mathcal{M}_0^{'}$ can be approximated by Equation (5.7), where the term $\alpha_\theta \frac{S_\theta}{2L} \left\|\mathbf{c}_k^1\right\|$ approximates the maximum possible distortion introduced by the quantization of θ_k (i.e. the distance between $\tilde{\mathbf{c}_k^1}$ and $\mathbf{c}_k^1$ in Figure 5.3), and the term $\alpha_r \frac{S_r}{2} \left\|\mathbf{e}_k^1\right\|$ is the maximum possible distortion introduced by the quantization of r_k (i.e. the distance between $\mathbf{c}_k^{1^{'}}$ and $\tilde{\mathbf{c}_k^1}$ in Figure 5.3).

$$D^{fr} \approx \sqrt{\left(\alpha_\theta \frac{S_\theta}{2L} \left\|\mathbf{c}_k^1\right\|\right)^2 + \left(\alpha_r \frac{S_r}{2} \left\|\mathbf{e}_k^1\right\|\right)^2}. \tag{5.7}$$

The minimum quantization steps S_θ^{min} and S_r^{min} that ensure the robustness of the fragile watermark against a vertex coordinate distortion of amplitude Dis can be obtained according to Equations (5.8) and (5.9).

$$S_\theta^{min} = Dis. \frac{2L}{\alpha_\theta \left\|\mathbf{c}_k^1\right\|}, \tag{5.8}$$

$$S_r^{min} = Dis. \frac{2}{\alpha_r \left\|\mathbf{e}_k^1\right\|}. \tag{5.9}$$

Similarity transformation and tolerable geometry compression can in fact be modeled as a slight vertex coordinate modification; thus, the quantization steps S_θ^{min} and S_r^{min} can be selected such that the embedded fragile watermark possesses a desired level of robustness against these tolerable operations, while being vulnerable to other non-tolerable modifications. In this way, these quantization steps are also usually small enough to ensure the watermark imperceptibility. The number of legal watermark symbols L is supposed to be large enough in order to ensure a small distortion (c.f. Equation (5.7)), a low false positive rate (c.f. the discussion in the previous subsection) and a high security level (e.g. to make it difficult to break out the symbol mapping mechanism). However, L cannot be too large due to the calculation and storage precision limitations

Table 5.2: Detailed information about the semi-regular meshes used in the experiments.

Semi-regular model ⇒	Venus	Rabbit	Horse	Feline
Maximum resolution level (J)	6	5	5	4
Edges in $\mathcal{M}_0$	491520	211968	337920	193536
Edges in $\mathcal{M}_J$	120	207	330	756

and also due to the fact that we have to ensure a desired robustness level to the tolerable operations. Once selected, these parameters can be fixed for almost all the semi-regular meshes without seriously affecting the algorithm's performances.

Our scheme can also be used as a high-capacity watermarking algorithm: watermark bits can be embedded independently in the quantized angle value $\theta_k^{'}$ and the quantized norm-length ratio $r_k^{'}$.

5.5 Experimental Results

5.5.1 Basic simulations

The proposed hierarchical watermarking framework has been implemented and tested on several semi-regular meshes. Figure 5.5 illustrates four of them: Venus, Rabbit, Horse and Feline. Table 5.2 lists some detailed information about these models. All the four meshes are obtained by using the remeshing technique of Guskov et al. [GVSS00] and are further normalized to within a unit sphere. Concerning the parameter setting, for the robust and the high-capacity watermarks, the control parameters ϵ_{rob} and ϵ_{hc} are fixed at 19 and 100 respectively, which appear to provide good performances for most of the models. After fixing ϵ_{rob}, for each mesh, we increase the DC factor α_{rob} as large as possible until visible distortion appears. α_{rob} values may also be fixed adaptively for different mesh regions according to their local geometric properties, such as the roughness measurement proposed in [Lav09]. This adaptive setting may lead to a better robustness and meanwhile a less induced perceptual distortion. The improvement on this point constitutes one part of our future work. For the fragile watermark, the used parameter values are as follows: $L = 32$, $S_\theta = \frac{1}{3}\pi = 60^\circ$, $S_r = 0.004$, $\alpha_\theta = 0.80$ and $\alpha_r = 0.99$.

Figure 5.6 illustrates the stego meshes in which three watermarks are embedded (under the same viewpoints as in Figure 5.5) and Figure 5.7 shows some close-ups of the watermarked and non-watermarked meshes. From these two figures, we can see that there exist nearly no perceptible distortions introduced by the watermark embedding, especially on relatively rough regions. However, on very smooth regions (e.g. the body of the Horse), some scar-like artifacts may appear, even under a very low watermarking

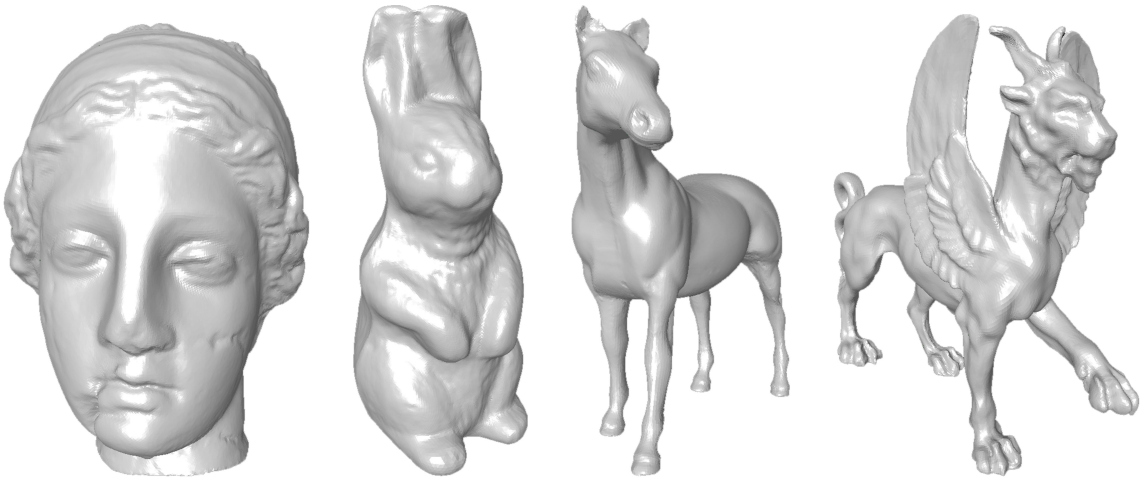


Figure 5.5: The original non-watermarked semi-regular meshes that are used in our experiments: (a) Venus, (b) Rabbit, (c) Horse, and (d) Feline.

strength. It is actually a common problem for 3-D mesh watermarking algorithms. The occurrence of this problem highlights the importance of the research on perceptual assessment of mesh watermarking algorithms [RAM06, LGD*06, CGEB07]. This kind of research may help to devise a perceptually adaptive watermark with different and appropriate strengths in different mesh spatial regions.

Table 5.3 lists the baseline evaluations of the hierarchical watermarking framework. All the tests have been carried out on a Pentium IV 2.8GHz processor with 2GB memory. The objective distortion between watermarked and original meshes is measured by Metro [CRS98] in terms of the maximum root mean square error (MRMS). A "perceptual" distance between them is evaluated by the mesh structural distortion measure (MSDM) proposed in [LGD*06] (with the radius parameter equal to 0.005, in order to have a relatively high sensitivity to the visual appearance variance). The MSDM value tends toward 1 (theoretical limit) when the measured objects are visually very different and is equal to 0 for identical ones. One advantage of the robust watermark is that it can introduce relatively high-amplitude objective modifications while keeping them perceptually invisible (the induced MSDM is always less than 0.15), since these modifications are rather of low frequencies. It is well known that for 3-D mesh watermarking, the lower frequency component modifications are both more imperceptible and more robust (c.f. Section 3.3.1.2). The MSDM remains low even after the embedding of all the three watermarks, which demonstrates the good imperceptibility of the whole hierarchical watermarking system. When comparing the distortions introduced by the different watermarks, we can observe that: 1) the induced distortion (either objective or

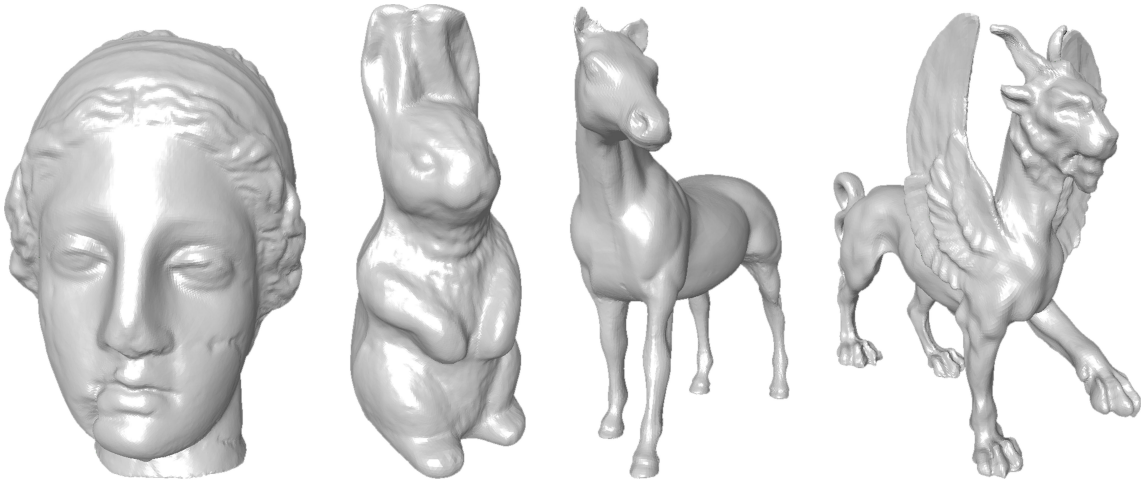


Figure 5.6: The watermarked semi-regular meshes: (a) Venus, (b) Rabbit, (c) Horse, and (d) Feline.

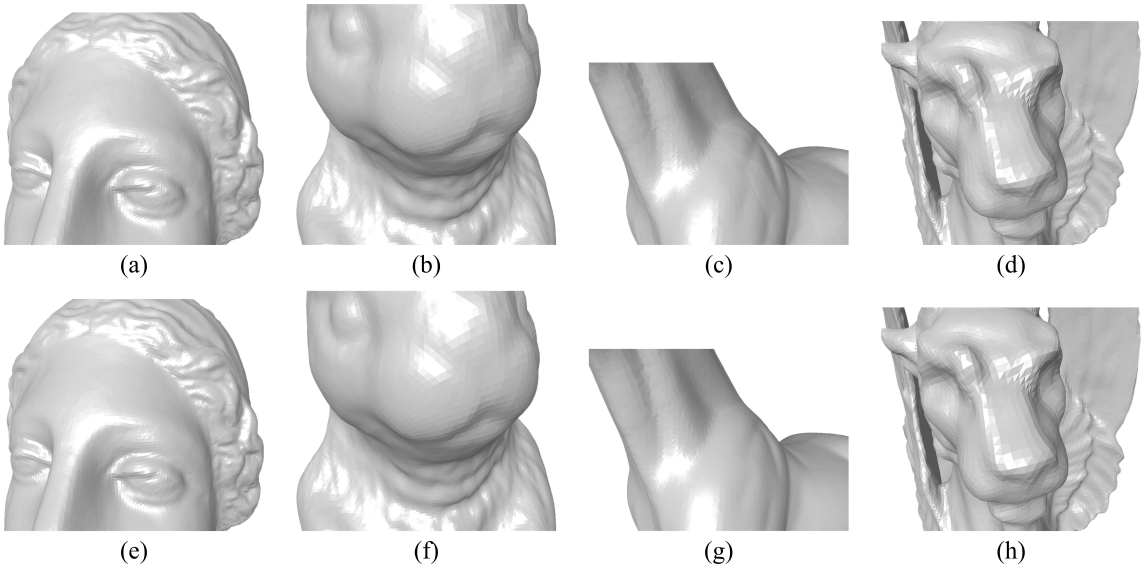


Figure 5.7: Some close-ups of the watermarked meshes: (a) Venus, (b) Rabbit, (c) Horse, and (d) Feline. The corresponding non-watermarked close-ups are also provided as (e)-(h) for comparison.

Table 5.3: Baseline evaluations of the hierarchical watermarking framework.

Semi-regular model ⇒	Venus	Rabbit	Horse	Feline
Embedding time of three WMs (s)	61.23	26.30	41.54	23.55
Extraction time of three WMs (s)	14.44	8.33	10.61	5.92
MRMS by three WMs (10^{-3})	1.24	1.15	0.67	0.72
MSDM by three WMs	0.085	0.103	0.13	0.12
Payload of the robust WM (bits)	64	64	64	64
Value of α_{rob}	0.50	0.70	0.50	0.85
Repetition time of the robust WM	1	3	5	1
MRMS by robust WM (10^{-3})	1.21	1.12	0.64	0.71
MSDM by robust WM	0.067	0.094	0.12	0.11
Embedding level of the H-C WM	4	4	4	3
Edges in that level	1920	828	1320	3024
Payload of the H-C WM (K bits)	7.632	3.18	5.247	11.925
MRMS by H-C WM (10^{-3})	0.22	0.20	0.15	0.12
MSDM by H-C WM	0.074	0.067	0.088	0.074
MRMS by fragile WM (10^{-3})	0.01	0.01	0.01	0.02
MSDM by fragile WM	0.049	0.046	0.059	0.050

"WM" stands for "watermark", and "H-C" stands for "high-capacity".

perceptual) of the robust watermark is much stronger than that due to the embedding of the high-capacity or fragile watermark; 2) the value of the overall distortion of the hierarchical watermarking framework is mainly determined by the distortion induced by the robust watermark insertion. Note that for the first three models, the maximum possible repetition rate is used for the 64-bit robust watermark. On the contrary, for the Feline model, although its has 756 edges in the coarsest-level, only the 64 longest edges are involved in the watermark embedding. Actually, the edges in the coarsest representation of Feline are very numerous so that they tend to have similar lengths and their associated WCVs tend to become of rather intermediate frequencies. Therefore, the watermark synchronization mechanism and the WCV norm quantization become less robust and the watermark repetition no longer improves the robustness. We believe that this will not induce ambiguity at extraction because we can easily estimate whether there exists watermark bit repetition or not by simply examining the autocorrelation function of the extracted bit sequence from all the coarsest-level edges.

Figure 5.8 illustrates the maps of the objective distortions introduced by different watermarks on the Rabbit mesh. The distortion pattern varies from relatively low frequency for robust watermark, to medium frequency for high-capacity watermark, and finally to high frequency for fragile watermark. Both the quantitative measurement results in Table 5.3 and the illustrations in Figure 5.8 confirm that the overall distortion of the hierarchical watermarking system is dominated by the robust watermark.

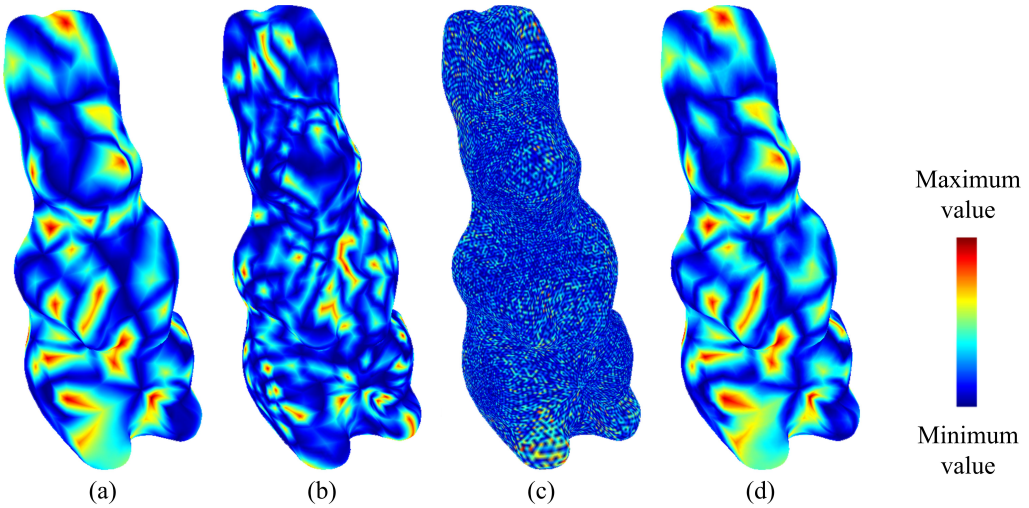


Figure 5.8: Maps of the objective distortions introduced by (a) the robust watermark, (b) the high-capacity watermark, (c) the fragile watermark, and (d) all the three watermarks on the Rabbit mesh. The minimum distance values of these four distortion maps are all zero. However, the maximum distance values are quite different: they are respectively 3.99×10^{-3}, 1.00×10^{-3}, 0.10×10^{-3} and 3.99×10^{-3} for the maps illustrated in (a), (b), (c) and (d).

5.5.2 Robust watermark test

The resistance of the robust watermark is tested under different geometry and file attacks, including similarity transformation, noise addition, smoothing, quantization and vertex/facet reordering. The robustness is measured by the normalized correlation [CMB*07] between the extracted watermark bit string $w_i^{'}, i \in \{1, 2, ..., M\}$ and the originally embedded string $w_i, i \in \{1, 2, ..., M\}$, as given by the following equation:

$$Corr = \frac{\sum_{i=1}^{M}(w_i^{'} - \bar{w^{'}})(w_i - \bar{w})}{\sqrt{\sum_{i=1}^{M}(w_i^{'} - \bar{w^{'}})^2 . \sum_{i=1}^{M}(w_i - \bar{w})^2}}, \tag{5.10}$$

where $\bar{w'}$ and $\bar{w}$ indicate respectively the averages of the watermark bit strings $w_i^{'}, i \in \{1, 2, ..., M\}$ and $w_i, i \in \{1, 2, ..., M\}$. This correlation value measures the similarity between two strings and varies between -1 (orthogonal strings) and $+1$ (the same strings).

In our tests, the maximum amplitude A of the additive noise is relative to the average distance from the vertices to the mesh center. The actual noise amplitudes on the vertex coordinates are pseudo-random values uniformly-distributed in the interval $[-A, A]$. For each amplitude level A, we perform five experiments using different seeds to generate different noise patterns and report the average as the final result. In smoothing attacks, the mesh is processed by Laplacian smoothing [Tauoo] with different iteration numbers while fixing the scaling factor λ as 0.10. In quantization attacks, the distance

from a vertex to the mesh center is quantized: an 8-bit quantization means that this distance is quantized to one of the 256 possible levels.

The robust watermark is experimentally invariant to both vertex/facet reordering and similarity transformation. Tables 5.4, 5.5 and 5.6 present respectively the evaluation results of the robustness against noise addition, smoothing and quantization. The attack-induced distortions, in terms of MRMS and MSDM, are provided in the third and fourth columns of these tables. Figure 5.9 illustrates several attacked Rabbit meshes. In general, our robust watermarking scheme works better on Venus, Rabbit and Horse than on Feline. The main reason is that the Feline has too many edges in its coarsest-level irregular mesh, thus the corresponding WCVs belong to rather intermediate frequencies and their modifications tend to be less robust. The second reason may be that the actual parameter setting is not very suitable for a mesh possessing so many edges in its coarsest-level representation.

More precisely, under noise addition, the performance of our scheme begins to decline when the MRMS error induced by an attack attains half of the MRMS distortion caused by the watermark embedding. This is reasonable because the watermark is embedded via a 2-symbol SCS quantization. The scheme shows better performance under smoothing and quantization than under noise addition for similar attack-induced MRMS errors. One explication is that the quantization step, which depends on the average edge length at the coarsest resolution level, has more chance to change consistently with the WCV norms under smoothing and quantization (i.e. both increase or both decrease) than under random noise addition, therefore the embedded watermark symbol have more chance to remain the same. It can also be observed from the tables that our robust watermark can withstand an attack that introduces a much higher MSDM distance (i.e. visual difference) than that caused by the watermark embedding. It is also important to note that the current parameter setting of our scheme is very conservative and in favor of the watermark imperceptibility and security instead of the robustness. A better robustness can be attained if we increase the watermark embedding strength.

Experimentally, the induced distortion increases as ϵ_{rob} decreases or α_{rob} increases. A perfect security is gained when $\alpha_{rob} = 0.50$, then the quantity of the leaked information increases as α_{rob} increases. One interesting point is the robustness variation along with the values of ϵ_{rob} and α_{rob}. When ϵ_{rob} is fixed, under a certain attack, it seems that there exists an optimum value for α_{rob} that maximizes the robustness. When α_{rob} is fixed, in general, the robustness is experimentally improved under small-amplitude attacks as ϵ_{rob} decreases. However, under moderate or strong attacks, robustness may not certainly be improved when ϵ_{rob} decreases. The main reason may be that the robustness is also

Table 5.4: Resistance of the robust watermark against random noise addition.

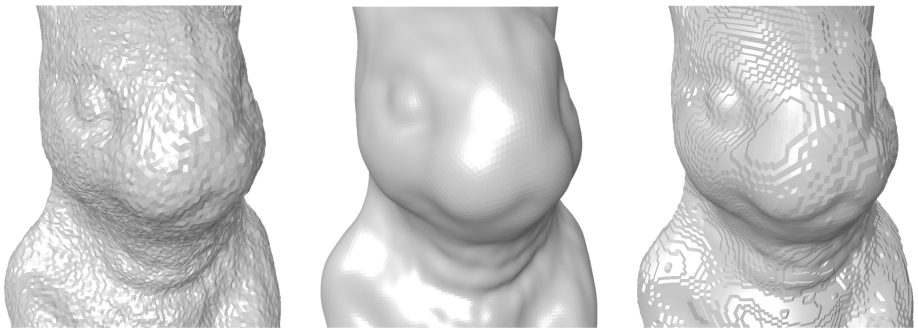
Model	Noise	MRMS (10^{-3})	MSDM	Correlation
Venus	0.05%	0.17	0.37	0.85
	0.25%	0.84	0.76	0.59
	0.50%	1.67	0.86	0.31
Rabbit	0.05%	0.11	0.27	0.92
	0.25%	0.55	0.70	0.59
	0.50%	1.10	0.83	0.31
Horse	0.05%	0.11	0.28	0.96
	0.25%	0.55	0.67	0.50
	0.50%	1.10	0.81	0.08
Feline	0.05%	0.13	0.19	0.78
	0.25%	0.63	0.55	0.39
	0.50%	1.26	0.70	0.02

Table 5.5: Resistance of the robust watermark against Laplacian smoothing ($\lambda = 0.10$).

Model	Iterations	MRMS (10^{-3})	MSDM	Correlation
Venus	10	0.27	0.18	0.74
	30	0.68	0.32	0.71
	50	1.01	0.39	0.62
Rabbit	10	0.24	0.19	0.90
	30	0.65	0.32	0.71
	50	1.03	0.39	0.45
Horse	10	0.21	0.16	0.97
	30	0.54	0.26	0.50
	50	0.80	0.30	0.35
Feline	5	0.33	0.13	0.74
	10	0.63	0.19	0.50
	30	1.59	0.32	-0.02

Table 5.6: Resistance of the robust watermark against coordinate quantization.

Model	Quantization	MRMS (10^{-3})	MSDM	Correlation
Venus	9-bit	0.93	0.57	0.93
	8-bit	1.85	0.72	0.70
	7-bit	3.70	0.83	0.63
Rabbit	9-bit	0.76	0.54	0.84
	8-bit	1.55	0.69	0.59
	7-bit	3.10	0.81	0.05
Horse	9-bit	0.68	0.49	0.61
	8-bit	1.35	0.64	0.25
	7-bit	2.70	0.76	0.17
Feline	10-bit	0.30	0.19	0.70
	9-bit	0.60	0.31	0.53
	8-bit	1.20	0.46	0.50



(a) (b) (c)

Figure 5.9: Some attacked Rabbit models for the robust watermark test: (a) by a 0.25% random noise addition, (b) by a Laplacian smoothing of 30 iterations ($\lambda = 0.10$), and (c) by an 8-bit quantization.

related to α_{rob} and to the error correction coding (here a redundant embedding).

5.5.3 ROC analysis of the robust watermark

As mentioned in Section 2.2, we can easily construct a *detectable* watermarking scheme from a *readable* scheme by adding a watermark presence decision step after the watermark extraction. In this subsection, we consider our robust watermarking algorithm as a detectable scheme and evaluate its performance in terms of receiver operating characteristics (ROC). The ROC can be analyzed either theoretically or experimentally, and with or without attack on the watermarked content. Basically, a ROC curve describes the relationship between the *false positive rate* against the *false negative rate* of the watermarking scheme under different threshold values for the watermark presence decision. The false positive rate represents the probability of the bad decision in which a watermark is detected but in fact it does not exist in the content. On the contrary, the false negative rate indicates the probability of the bad decision in which a watermark is not detected but in fact it exists in the content.

The ROC performance of our detectable robust watermarking algorithm under noise addition and Laplacian smoothing attacks have been experimentally analyzed. To complete the ROC curves, we first prepared 100 watermarked meshes of the same object (Rabbit) using different random watermarks and random keys. The algorithm parameter values are the same as mentioned previously. We then attacked these models through noise addition or Laplacian smoothing (with different amplitudes); for each attacked model, two detections were performed: one with the right watermark and the right key, and the other with a wrong watermark and a wrong key. Then, for each kind of attack (noise addition or smoothing, with different amplitudes), the false positive and false

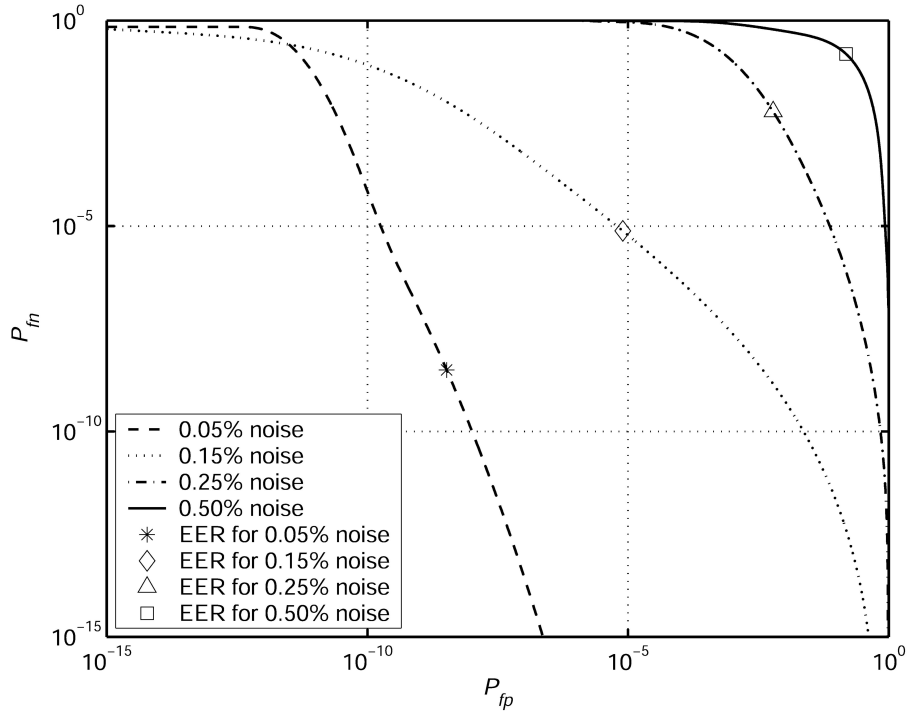
negative curves are drawn by varying the correlation threshold value that is used to decide the watermark presence. These curves are approximated to Gaussian distributions and the ROC curves that represent the relationship between the false negative value P_{fn} and the false positive value P_{fp} are therefore obtained. According to the experimental results illustrated in Figure 5.10 where the equal error rates (EER) of the curves are also indicated, our method demonstrates satisfactory performances under both kinds of attacks, even with relatively strong amplitudes. For instance, under a 0.15% random noise addition, an appropriate threshold value can be found so that the false positive and the false negative probabilities are both approximately equal to 10^{-5}.

5.5.4 High-capacity watermark test

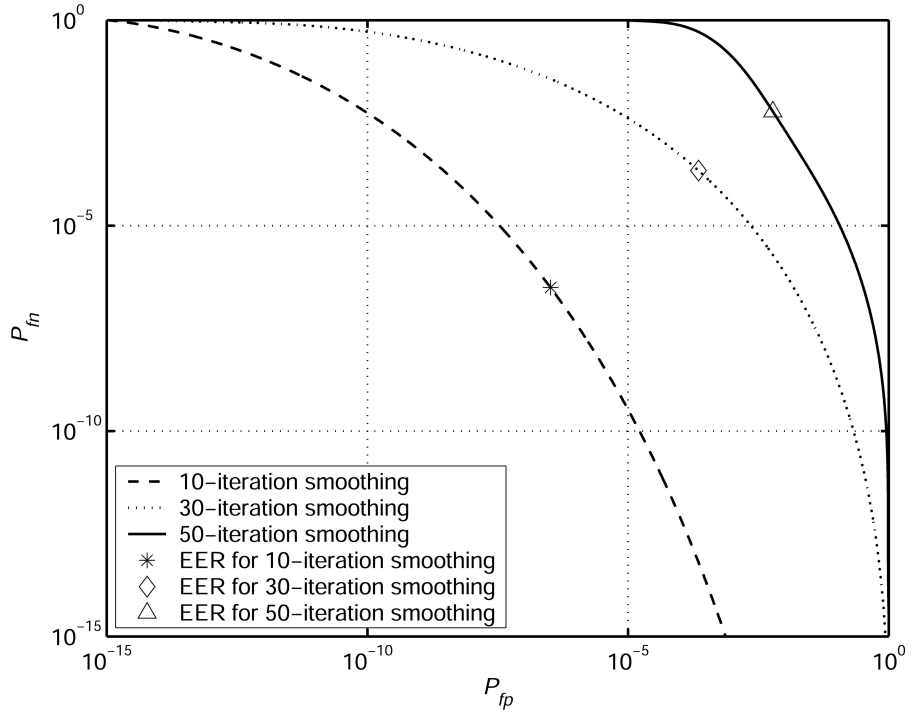
As anticipated, the high-capacity watermark is robust against vertex/facet reordering and similarity transformation, but somewhat fragile to the other attacks. It can resist until about 0.002% to 0.004% random additive noise. The maximum resistible noise amplitude seems inversely proportional to the edge number of the resolution level where the high-capacity watermark is embedded. The imperceptibility of the high-capacity watermark mainly depends on the value of ϵ_{hc}: when ϵ_{hc} decreases, the watermark becomes more visible. For example, when the watermark is embedded at level 4 of Rabbit, the critical value for ϵ_{hc} is about 70 beyond which the watermark becomes visible. The watermark payload depends on the parameter G, which has an upper limit due to the limited vertex coordinate storage precision. This precision is fixed as 6 digits per coordinate in our experiments. If the watermark is embedded at level 4 of Rabbit with $\epsilon_{hc} = 70$, G can be increased up to the total edge number of this level (828 edges), leading to a payload of 6.837K bits. On the contrary, if the watermark is embedded at level 3 that has 3312 edges, with $\epsilon_{hc} = 70$, G cannot be greater than 750 (then the payload is 24.34K bits) in order to ensure the correctness of the embedded watermark under a 6-digit vertex coordinate storage precision.

5.5.5 Fragile watermark test

We have fixed the parameters of the fragile watermark so that it can resist an angle distortion until about 1° and a WCV norm distortion until about 0.2% of the minimum edge length (c.f. Equations (5.8) and (5.9)). In order to verify its effectiveness, several attacked models have been prepared. These attacks include similarity transformation, local invisible noise addition, local deformation, local rotation and global geometry processing. Figure 5.11.(a)-(e) illustrates the attacked Rabbit models. Their corresponding



(a)



(b)

Figure 5.10: ROC curves of the detectable robust watermarking algorithm under (a) random noise addition and (b) Laplacian smoothing ($\lambda = 0.10$). The tests have been carried out on the Rabbit model.

authentication results are shown in Figure 5.11.(f)-(j). The watermark is experimentally invariant to similarity transformation (c.f. Figure 5.11.(f)). According to the watermark extraction results, we can successfully locate the noised part (c.f. Figure 5.11.(g)) and the deformed part (c.f. Figure 5.11.(h)) on the modified models. We can also report that the watermarked model may have undergone a local rotation (c.f. Figure 5.11.(i) where the neck of the Rabbit is invalid since its head has been rotated). Finally, we can also detect a global geometry modification, such as a smoothing in Figure 5.11.(j).

5.6 Conclusion

In this chapter, a hierarchical and multiple watermarking framework for semi-regular meshes has been proposed. Three different watermarks (robust, high-capacity and fragile) can co-exist in a same semi-regular mesh, serving for different applications (copyright protection, content enhancement and content authentication). The contributions of this work can be summarized as follows:

- First, as far as we know, the proposed hierarchical watermarking framework constitutes the first attempt on multiple mesh watermarking in the literature;
- Second, the robust watermark demonstrates a relatively good performance, mainly due to the selection of an appropriate watermarking primitive and the adoption of a robust watermark synchronization mechanism;
- Third, to the best of our knowledge, the fragile watermark is the first on this topic that is robust against all the content-preserving operations while providing a precise attack localization capability;
- Fourth, the high-capacity watermark combines the basic ideas of the geometry-based and order-based schemes and thus possesses the good properties of both kinds of techniques;
- Finally, during the algorithms' design, we have explicitly taken the watermarking security into account.

In the next two chapters, we will focus on the blind and robust watermarking of general meshes that can resist the intractable connectivity attacks. Chapter 6 will describe a spatial method while in Chapter 7 we will present a spectral method.

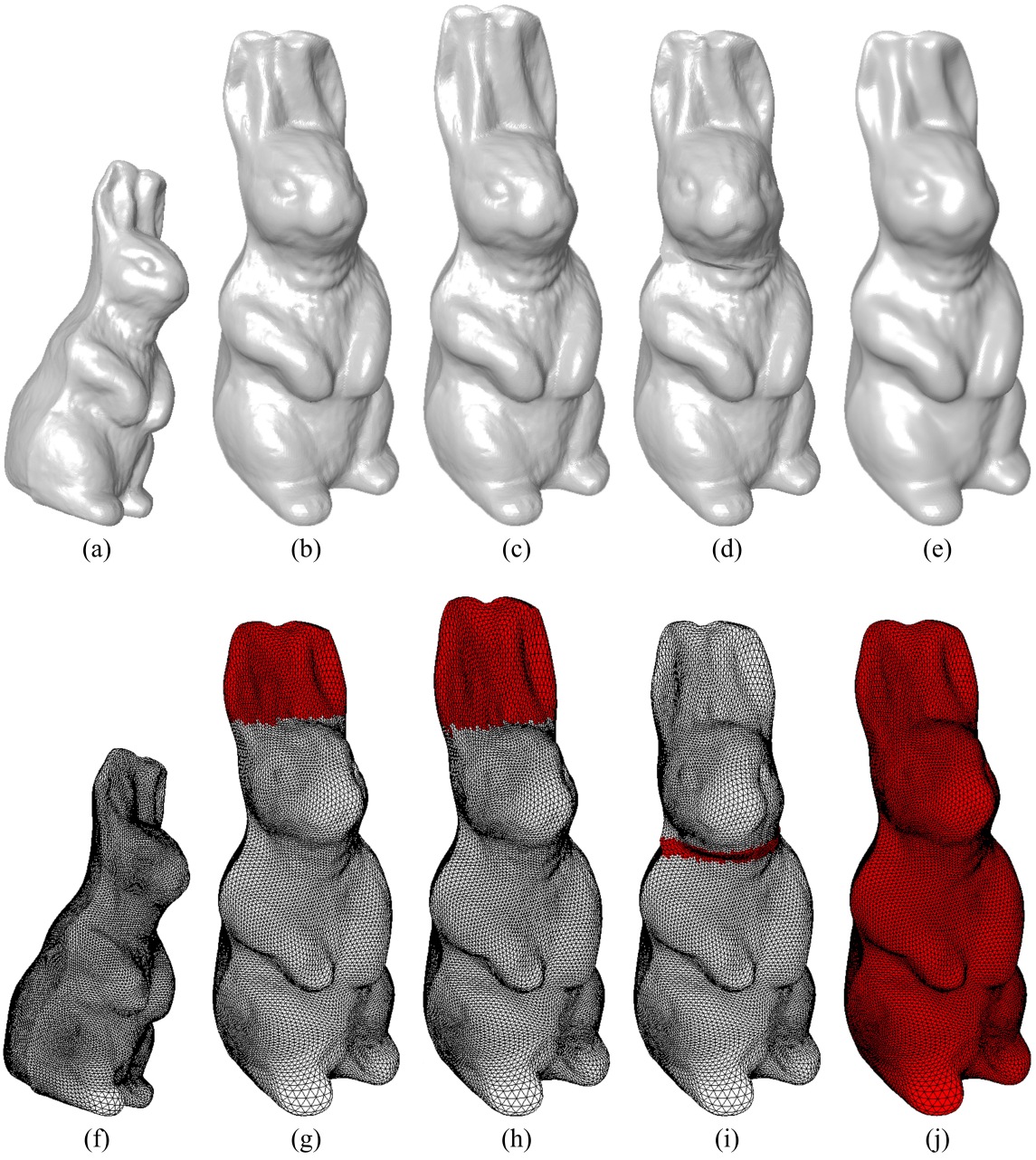


Figure 5.11: Some attacked Rabbit models for the fragile watermark test: (a) by a similarity transformation, (b) by a 0.0005% binary invisible noise on the ears, (c) by a local deformation where the ears have been pulled up, (d) by a local rotation where the head has been rotated for 15°, and (e) by a 10-iteration Laplacian smoothing with $\lambda = 0.10$. The corresponding authentication results are visualized in (f)-(j), where the valid parts are rendered in white, while the invalid parts are rendered in red.

Chapter 6

Robust and Blind Mesh Watermarking Based on Volume Moments

Contents

In this chapter, we present a robust and blind mesh watermarking method. The selected watermarking primitive is an intrinsic 3-D shape descriptor: the analytic and continuous geometric volume moment. During the watermark embedding, the cover mesh is first normalized to a canonical and robust spatial pose by using its global volume moments. Then, the normalized mesh is decomposed into patches and a multi-bit watermark is embedded through a modified SCS quantization of the zero-order volume moments of some selected candidate patches. Experimental results and comparisons with the state of the art demonstrate the superiority of the proposed approach in terms of robustness and imperceptibility. Moreover, to the best of our knowledge, our method is the first in the literature that addresses the robustness against 3-D shape representation conversions (e.g. discretization of the stego mesh into voxels).

A paper describing the proposed scheme has been submitted to an international journal [WLDB09b] and is currently under major revisions.

6.1 Introduction and Basic Idea

This chapter focuses on the robust and blind watermarking of 3-D meshes. The design of such algorithms is considered to be very difficult mainly due to the mesh irregular sampling nature and the existence of many intractable attacks [WLDB08a] (c.f. Section 3.1). Besides the geometry attacks that only modify the vertex positions, the connectivity attacks can completely change the geometry and the connectivity information of the watermarked mesh while well conserving its global shape. The most destructive attack may be the 3-D object representation conversion (e.g. from mesh to voxels); the mesh itself disappears after such a conversion. These attacks are actually common operations in various mesh applications. When performing these operations, normally the user also tries to preserve the basic shape (i.e. the visual appearance) of the model. Indeed, usually a too much distorted object does not present too much interest to the user and the application. Following this idea, we believe that a valuable robust mesh watermark has to be linked to the *3-D shape* that is behind the mesh, but not to the mesh itself. Hence, we have chosen an intrinsic 3-D shape descriptor, i.e. the *geometric volume moment*, as the watermarking primitive. This descriptor is of continuous nature and depends only on the 3-D analytic shape represented by the mesh. Therefore, it should be robust against geometry, connectivity and representation conversion attacks providing that they do not seriously modify the shape of the stego object. In our method, a multi-bit blind watermark is embedded in the cover mesh by slightly modifying its local geometric volume moments through a modified and "adaptive" SCS quantization.

In fact, the watermarking primitives used in the methods of Cho et al. [CPJ07] and Zafeiriou et al. [ZTP05] (which are deemed to be the most effective blind mesh watermarking schemes proposed so far) are statistical mesh shape descriptors. In their methods, the watermark is embedded through distribution modification of a geometric histogram that is derived from the vertex coordinates. The histograms used in these two algorithms represent more or less the intrinsic properties of the mesh shape. However, they are still of discrete nature since their construction depends entirely on the coordinates of the mesh vertices, which only constitute a discrete sampling of the continuous mesh surface. The consequence is that their watermarking schemes are not robust against (spatially) anisotropic connectivity attacks because these attacks will seriously alternate the distribution of the histograms. On the contrary, the geometric volume moment is of continuous nature and as shown in the following, it possesses a very high robustness against various attacks (even anisotropic) as long as the mesh shape is not seriously modified.

Another critical issue for 3-D mesh blind watermarking is the causality problem, which means that the posteriorly embedded watermark signal components disturb the correctness and/or the synchronization of the previously embedded ones (c.f. Section 3.2.1.1). For instance, in the method of Bors [Boro6], the author establishes an order for the watermarking candidate vertices according to a geometric criterion, and then modifies another correlated geometric metric to embed watermark bits. The established vertex order may be altered after the watermark bit embedding; in consequence, the author has to introduce a post-processing step to recover the original vertex order, so as to ensure the correctness of the watermark extraction when there is no attack. In our algorithm, after watermark embedding, we introduce a geometric *compensation* process in order to resolve the causality problem. This compensation process recovers some initial mesh features that are critical to the watermarking algorithm.

Another important point that has to be taken into account is the watermark imperceptibility. It has been demonstrated that the watermark embedding in the mesh low-frequency components can be both more robust and more imperceptible [SCOT03, ZvKD07]. We have followed this principle when devising our method.

Hence, we present a new robust and blind mesh watermarking algorithm based on the analytic and continuous geometric volume moments of the mesh object. The global moments are used to perform a robust mesh registration preprocessing before watermark embedding, and the local moments are used as the watermarking primitives. The remainder of this chapter is organized as follows: Section 6.2 introduces the geometric volume moments; Section 6.3 provides an overview of the proposed method; Sections 6.4 and 6.5 detail respectively the watermark embedding and extraction procedures; Section 6.6 presents some experimental results; finally Section 6.7 concludes the chapter.

6.2 Geometric Volume Moments

For a closed 3-D surface $\mathcal{S}$, its geometric volume moments (of different orders) are defined according to the following equation:

$$m_{pqr} = \int\int\int x^p y^q z^r \rho\left(x, y, z\right) dxdydz, \tag{6.1}$$

where p, q, r are the orders, and $\rho\left(x, y, z\right)$ is the volume indicator function (it is equal to 1 if (x, y, z) is inside the closed surface $\mathcal{S}$; otherwise it is equal to 0). The volume moment of order p, q, r is actually the (continuous) volume integration of the function $f\left(x, y, z\right) = x^p y^q z^r$ inside the closed surface $\mathcal{S}$.

For an orientable 3-D polygonal mesh, Zhang and Chen [ZC01] and Tuzikov et al.

[ST01, TSV03] derived independently the explicit expression for the volume integration given by Equation (6.1). The basic idea is to calculate it as a sum of signed integrations over several elementary volumes. For a triangular mesh, the elementary volume is the tetrahedron constituted of a triangle facet f_i and the coordinate system origin $\mathcal{O}$. The contribution sign for each tetrahedron is determined according to the orientation of f_i and the relative position between f_i and $\mathcal{O}$. Note that if the facets are correctly oriented (i.e. the normals of the facets all point to the outside of the closed surface), then the zero-order moment m_{000} is the volume of the closed surface. Some of the low-order elementary moment integration expressions $m_{pqr}^{(f_i)}$ are listed as Equations (6.2) to (6.5), where $f_i = \{v_{i1}, v_{i2}, v_{i3}\} = \{(x_{i1}, y_{i1}, z_{i1}), (x_{i2}, y_{i2}, z_{i2}), (x_{i3}, y_{i3}, z_{i3})\}$. A more complete list of the elementary moment calculation expressions can be found in the papers of Tuzikov et al. [ST01, TSV03].

$$m_{000}^{(f_i)} = \frac{1}{6}\left|x_{i1}y_{i2}z_{i3} - x_{i1}y_{i3}z_{i2} - y_{i1}x_{i2}z_{i3} + y_{i1}x_{i3}z_{i2} + z_{i1}x_{i2}y_{i3} - z_{i1}x_{i3}y_{i2}\right|, \tag{6.2}$$

$$m_{100}^{(f_i)} = \frac{1}{4}\left(x_{i1} + x_{i2} + x_{i3}\right) m_{000}^{(f_i)}, \tag{6.3}$$

$$m_{200}^{(f_i)} = \frac{1}{10}\left(x_{i1}^2 + x_{i2}^2 + x_{i3}^2 + x_{i1}x_{i2} + x_{i1}x_{i3} + x_{i2}x_{i3}\right) m_{000}^{(f_i)}, \tag{6.4}$$

$$m_{110}^{(f_i)} = \frac{1}{10}\left(x_{i1}y_{i1} + x_{i2}y_{i2} + x_{i3}y_{i3} + \frac{x_{i1}y_{i2} + x_{i1}y_{i3} + x_{i2}y_{i1} + x_{i2}y_{i3} + x_{i3}y_{i1} + x_{i3}y_{i2}}{2}\right) m_{000}^{(f_i)}. \tag{6.5}$$

With the above calculation, geometric volume moments can be easily generalized to non-closed orientable surfaces (e.g. a mesh patch). The calculation consists in first adding fictional facets by connecting the boundary vertices and the origin, and then calculating the moments of the obtained closed surface. These geometric moments are very robust features and have been used for mesh self-registration and in 3-D shape retrieval [ZC01]. In this chapter, we will use the global volume moments for mesh normalization (as a preprocessing before both watermark embedding and extraction) and the local volume moments as the watermarking primitives. In fact, in the case of 2-D images, invariant and orthogonal moments have already been used for constructing robust watermarking methods in [KL03], [AT04] and [XLP07].

6.3 Overview of the Proposed Method

As mentioned in Section 6.1, the proposed watermarking method is based on the assumption that a good mesh watermarking primitive has to be intrinsically linked to the

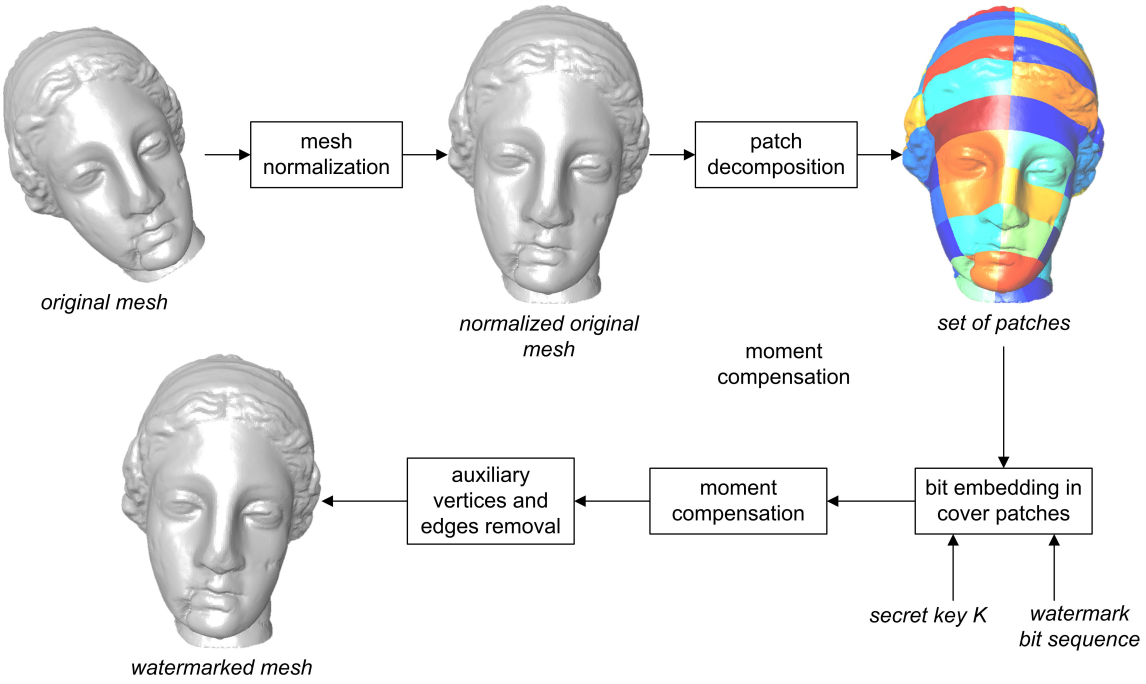


Figure 6.1: Block diagram of the watermark embedding procedure.

3-D shape that is behind the mesh. The analytic and continuous volume moments presented in the last section seem to be very good candidates. We wish to consider them as primitives to embed a multi-bit readable watermark (in contrast to a detectable watermark) in a cover mesh. We immediately encounter two difficulties: first, the volume moments of different orders are correlated so it becomes very complicated to modify different moments of a same mesh simultaneously and independently (in order to embed multiple bits); second, there does not exist an easy and straightforward way to transform the modifications of the volume moments (so as to embed watermark bits) back into the spatial domain of the cover mesh, because we cannot easily deduce the mesh vertex coordinates that produce the volume moments of some prescribed values. The first point forces us to decompose the mesh into several patches and embed one bit in each patch. For the second point, we devise an iterative patch deformation algorithm which is quite effective in sense that the target patch moment value can normally be attained within just few iteration steps.

Figure 6.1 illustrates the bloc diagram of our watermark embedding procedure. The cover mesh is first normalized by using its global volume moments. Then, the mesh is transformed from Cartesian coordinate system (x, y, z) to cylindric coordinate system (h, r, θ). Afterward, we decompose the normalized mesh into patches by discretizing the obtained h and θ domains. For several selected patches (the *cover* patches), we calculate their zero-order moments and quantize them so as to embed one bit per patch. Note

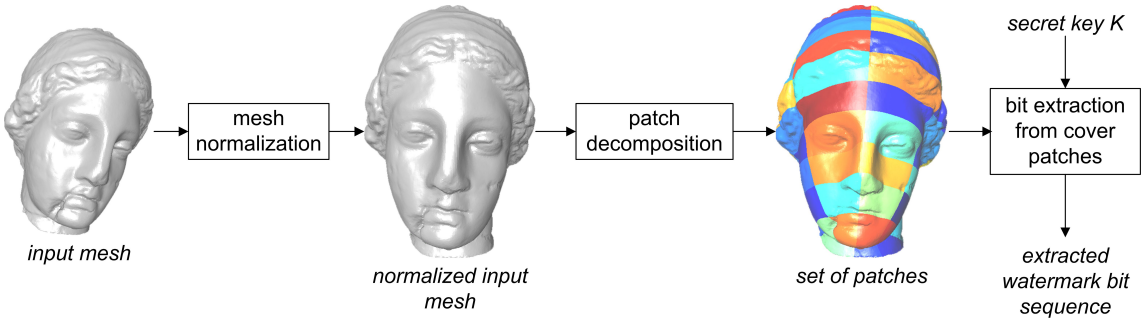


Figure 6.2: Block diagram of the watermark extraction procedure.

that in order to ensure a precise patch moment calculation, we insert some auxiliary vertices and edges on the patch borders; they can be easily removed after the watermark embedding. The moment modification is carried out through iterative patch deformation. In order to keep the watermark induced distortion as imperceptible as possible, we integrate a smooth modulation mask into the iterative patch deformation process. The causality problem arises at this point, because after the deformation of the cover patches, the mesh global volume moments are probably changed so that we cannot achieve the same normalized mesh pose at extraction in a blind way. A moment compensation post-processing step is introduced to resolve this causality problem.

Figure 6.2 illustrates the watermark extraction procedure. It does not require the original non-watermarked mesh and only needs a secret key with the objective to ensure the watermarking security. The extraction consists of three steps: mesh normalization, patch decomposition and watermark bits extraction from the obtained cover patches. In the next two sections, the watermark embedding and extraction algorithms will be described in details.

6.4 Watermark Embedding

In this section, we present the different steps of the watermark embedding procedure that is illustrated in Figure 6.1.

6.4.1 Mesh normalization

Mesh normalization is used as a preprocessing step by both watermark embedding and watermark extraction, and consists of three sequential operations:

1. Translation of the mesh so that its center coincides with the origin of the objective Cartesian coordinate system;
2. Uniform scaling of the mesh so that it is bounded within a unit sphere;

3. Rotation of the mesh so that its three principal axes coincide with the axes of the objective coordinate system.

The mesh center coordinates are calculated as the following ratios between the volume moments of different orders [ZC01]:

$$C = (x_c, y_c, z_c) = \left(\frac{m_{100}}{m_{000}}, \frac{m_{010}}{m_{000}}, \frac{m_{001}}{m_{000}} \right). \tag{6.6}$$

The principal axes of the mesh are computed as the ordered eigenvectors (according to their associated eigenvalues) of the following matrix [ZC01]:

$$M = \begin{bmatrix} m_{200} & m_{110} & m_{101} \\ m_{110} & m_{020} & m_{011} \\ m_{101} & m_{011} & m_{002} \end{bmatrix}. \tag{6.7}$$

In our implementation, the most significant principal axis is aligned with the axis Z. In order to resolve the axis alignment ambiguity problem, besides the compliance to the right-hand rule of the three principle axes, we impose two other geometric constraints (i.e. the global moments m_{300} and m_{030} of the rotated mesh should be positive, as suggested in [ZC01]). In this way, we ensure that the obtained aligned object is unique and consistent. Note that the volume moments m_{100}, m_{010}, m_{001}, m_{110}, m_{101} and m_{011} of the normalized mesh are all equal to zero.

The above normalization relies on the analytic volume moments and therefore is processed in a continuous space. So far, in most existing watermarking methods, the mesh normalization step depends entirely on the vertex coordinates, while completely discarding the mesh connectivity information [KTP03, ZTP05, CPJ07]. This kind of "*discrete*" moment is not very robust, especially against anisotropic connectivity attacks. Recently, Rondao-Alface et al. [RAMC07] have calculated the mesh center as the weighted average coordinates of the vertices, which is somewhat equivalent to the calculation based on the mesh *surface* moments [TSV03]. Table 6.1 compares the robustness of the mesh normalizations based on discrete, surface and volume moments, in terms of the center norm variation $V_{\|C\|}$ and the maximum principal axis variation MV_{PA} (in degrees). The experiments were performed on the Venus mesh (100759 vertices) that is illustrated in Figures 6.1 and 6.2. The tested attacks include noise addition (spatially uniform or anisotropic), vertex coordinate quantization and surface simplification (spatially uniform or anisotropic). The conducted attacks in our experiments are considered to have very strong amplitudes in the context of mesh watermarking (c.f. Figure 6.10, the visual effects of such attacks can be easily perceived). From Table 6.1, it can be observed

Table 6.1: Robustness comparison of the different mesh normalization schemes on the Venus model under various strong-amplitude attacks.

Attack	Discrete moments		Surface moments		Volume moments	
	$V_{\|\|C\|\|}$	MV_{PA}	$V_{\|\|C\|\|}$	MV_{PA}	$V_{\|\|C\|\|}$	MV_{PA}
0.50% noise	1.12×10^{-6}	0.003°	6.36×10^{-4}	0.23°	3.74×10^{-5}	0.01°
7-bit quantization	1.57×10^{-5}	0.01°	2.98×10^{-3}	1.07°	2.70×10^{-5}	0.05°
90% simplification	3.29×10^{-3}	3.32°	3.95×10^{-4}	0.05°	1.18×10^{-4}	0.03°
0.50% anisotropic noise	8.02×10^{-6}	0.01°	0.044	5.70°	2.39×10^{-5}	0.01°
50% anisotropic simplification	0.40	82.53°	2.30×10^{-3}	0.18°	5.51×10^{-4}	0.05°

that the mesh normalization based on the volume moments possesses the best overall performance, especially under spatially anisotropic noise addition and simplification.

6.4.2 Decomposing the mesh into patches

The normalized mesh is decomposed into several patches in order to make it possible to embed a multi-bit watermark. Indeed, one bit is embedded in each selected candidate patch among the patches that are generated by this decomposition step.

First, the original Cartesian vertex coordinates $v_k = (x_k, y_k, z_k)$ are converted into the cylindric coordinate system as

$$v_k = (h_k, r_k, \theta_k) = \left(z_k, \sqrt{x_k^2 + y_k^2}, tan^{-1}\left(\frac{y_k}{x_k}\right)\right). \tag{6.8}$$

Then, the mesh is segmented into several patches through a simple uniform discretization of the h and θ domains into I_h and I_θ intervals. We denote the two discretization steps by h_{step} and θ_{step} (respectively for the h and the θ domains). This discretization may be pseudo-randomized by introducing a secret key that determines the values of the h and θ domain ranges of the generated patches, with the objective to further enhance the watermarking security. However, for the sake of simplicity, we adopt the straightforward uniform discretization in our experiments.

Each mesh vertex is associated to its proper patch according to its discretized indices denoted by $ind(h_k)$ and $ind(\theta_k)$; however, there exist some facets that cover multiple patches. These facets have to be split into several small ones, each of which completely lies in a single patch. This facet split process is necessary so as to ensure a precise patch moment calculation, which is critical to the watermark robustness. The facet splitting process is accomplished by automatically adding auxiliary vertices and edges on the patch borders (c.f. Figure 6.3.(c)-(d)). The whole decomposition process can be considered as a segmentation of the mesh by intersecting some 3-D planes with the mesh surface in a continuous space. The mesh is thus decomposed into $I_h \times I_\theta$ patches $\mathcal{P}_j, j \in \{0, 1, ..., I_h.I_\theta - 1\}$. These patches are ordered according to their spatial positions

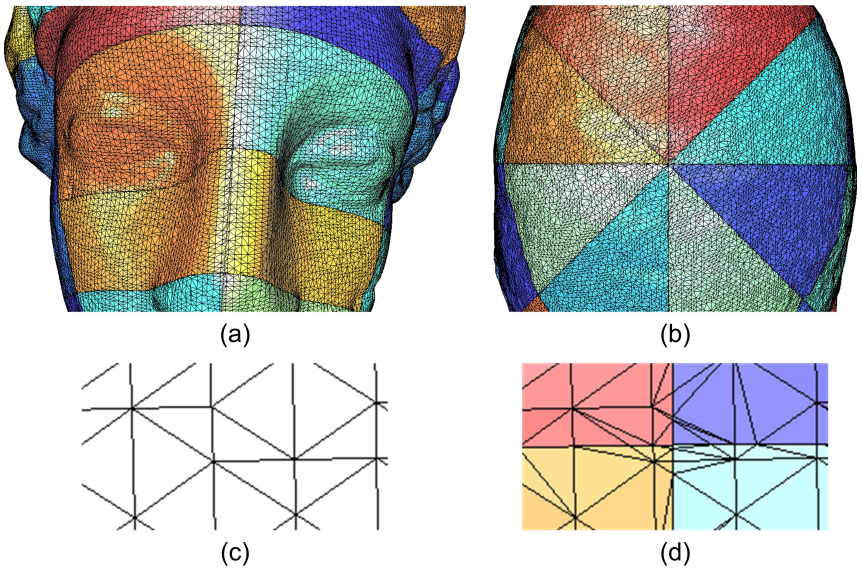


Figure 6.3: In this figure, (a), (b) and (d) illustrate three close-ups of the decomposed Venus mesh; the original connectivity of (d) is shown in (c).

and their indices are determined as $j = ind(h_k).I_\theta + ind(\theta_k)$.

I_h and I_θ are two important parameters of our watermarking algorithm: if we increase the patch number $I_h \times I_\theta$, then the watermarking capacity will be increased; but meanwhile this will experimentally induce higher-amplitude patch deformation if a comparable robustness level is required, and visible distortions are thus prone to occur. The explanation is as follows: when the mesh is decomposed into a large number of patches (imagine the extreme case where each patch contains just one vertex), the local deformation of the patches will become of high frequency, which has been proven to be more visible and less robust [SCOT03, ZvKD07]. Experimentally, the parameter setting $I_h = 11$ and $I_\theta = 8$ seems to achieve a good trade-off between watermark capacity, robustness and imperceptibility for most meshes. Moreover, it can normally ensure a watermark capacity of around 64 bits, which is a common and sufficient payload for a robust readable watermark used in copyright protection applications [KP99]. An adaptable setting of these two parameters according to the specific shape of the cover mesh constitutes one part of our future work.

The combination "mesh normalization + cylindric discretization" constitutes a simple but effective mesh decomposition process. First, it can reproduce exactly the same decomposition at extraction in a blind way, with an intrinsic patch order. Besides, this decomposition depends only on the center and the principal axes of the object and is also very robust against various geometry and connectivity attacks (as proven by the experimental results given in Table 6.1 and also those that will be presented in the next

paragraph). Also note that even if the adopted cylindric decomposition is not one-to-one (e.g. two different vertices in the object may have the same $h - \theta$ values), the mesh can still be robustly and consistently decomposed as long as it is orientable; indeed we do not want to create a real mapping such as in mesh parametrization [FH05].

The robustness of this decomposition (i.e. normalization + discretization) has been tested by analyzing the stability of the patch zero-order volume moment values under various attacks, even those that are spatially non-uniform (i.e. anisotropic). Figure 6.4 presents the experimental results on the watermarked Horse (112642 vertices) that is illustrated in Figure 6.8.(b). It can be seen that the curves of the patch moment values almost coincide under the studied strong-amplitude attacks. These results demonstrate the robustness of the patch decomposition, the stability of the patch moment values, and also the interest of using these local volume moments as the watermarking primitives. The proposed mesh decomposition method is not robust against strong local deformation and cropping, which are quite difficult to handle for blind mesh watermarking methods. These attacks cause serious desynchronization problems (i.e. the generated patches in which watermark bits are embedded are not consistent) due to the deviation of the mesh normalization process. Our normalization also fails for spheres and some other special objects, for which it is difficult to estimate the principal axes; however, most existing mesh segmentation methods would also fail to decompose consistently a sphere object. Moreover, in real life, this kind of n-symmetric object remains marginal.

6.4.3 Patch classification and watermark synchronization

The obtained patches are classified into three groups:

1. *cover* patches for watermark bit embedding;
2. *discarded* patches not suitable for deformation;
3. *compensation* patches for moment compensation after watermark bit embedding.

The discarded patches will not be used for bit embedding or for moment compensation. They are actually some small patches having either a very low zero-order volume moment amplitude, or a very small h domain range, or a very small θ domain range. It is in practice very difficult to deform these singular patches equally strongly as the other patches, and their volume moments are not that robust compared to the other patches; therefore they are discarded and will not be deformed in our watermark embedding algorithm. Three empirical thresholds $\bar{m_{000}} = 0.0005$ for zero-order volume moment amplitude, $\bar{h_r} = 0.35 \times h_{step}$ for h domain range, and $\bar{\theta_r} = 0.35 \times \theta_{step}$ for θ domain range are established to filter out these patches.

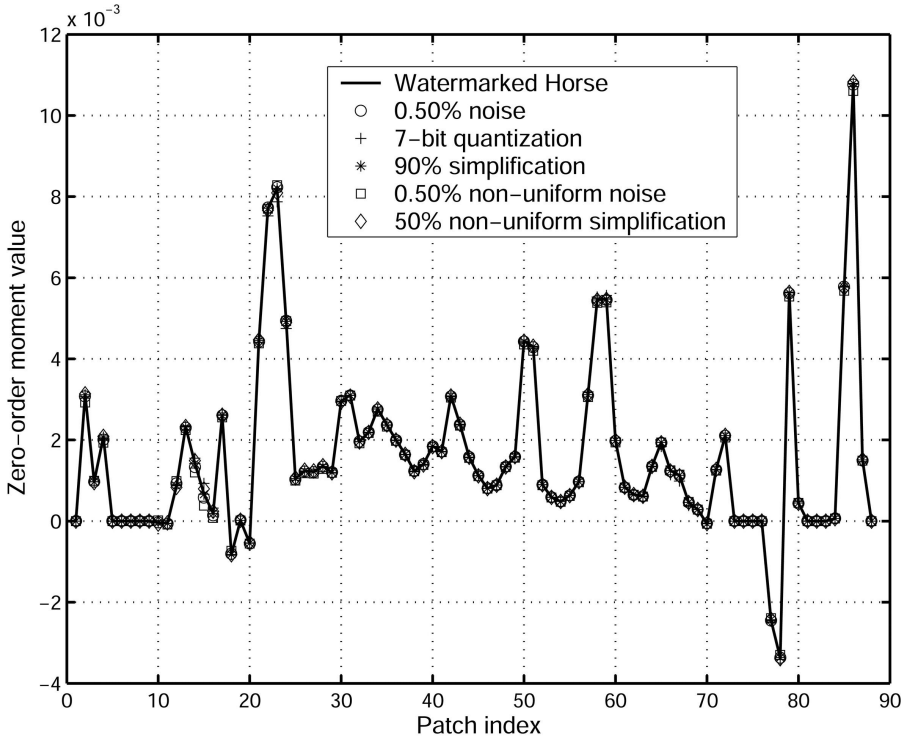


Figure 6.4: Stability of the patch moment values of the watermarked Horse under various strong-amplitude attacks.

The compensation patches serve to be deformed after watermark bit embedding in the cover patches, so as to recover the mesh center position and principal axis orientations. This recovery of the mesh canonical pose is necessary to prevent the causality problem and thus critical to the correctness of the watermark embedding. The patches with larger moment amplitudes are favorable for this task since they allow a larger moment variation while keeping the deformation imperceptible. The 12 patches with the largest m_{000} amplitudes are kept from watermark bit embedding and considered as compensation patches. They are denoted by $\mathcal{P}_l^c, l \in \{0, 1, .., 11\}$. A compensation patch with a smaller index in this sequence has a larger m_{000} amplitude.

All the other N patches are cover patches and denoted by $\mathcal{P}_n^w, n \in \{0, 1, .., N-1\}$. A cover patch with a smaller index in this sequence also has a smaller index in the global indexing $\mathcal{P}_j, j \in \{0, 1, ..., I_h.I_\theta - 1\}$. This cover patch order is used for the watermark synchronization: watermark bits are sequentially embedded in or extracted from these ordered cover patches.

The above patch classification may introduce causality and desynchronization problems. For instance, after the watermark embedding or an attack, a compensation patch may become a cover patch if its m_{000} amplitude decreases. In order to prevent these problems from happening, we take out some special measures concerning the poten-

tially sensitive patches with regard to the patch classification. For example, the m_{000} amplitude of the compensation patch $\mathcal{P}_{11}^{c}$ is constrained to be increased during moment compensation, so as to prevent it from being classified as a cover patch after compensation process and also to decrease the probability for it to become a cover patch after an attack. After taking out these measures, the original patch classification is preserved and enhanced. Experimentally, watermark desynchronization never happens under weak and moderate attacks, and also rarely occurs under strong attacks. This desynchronization problem can also be resolved by transmitting an additional sequence of $I_h \times I_\theta$ bits to the watermark extraction side in order to explicitly indicate the locations of the cover patches. Since this supplementary information is of small amount (i.e. less than 128 bits), we still consider the corresponding watermark extraction as a blind algorithm (c.f. Section 2.2).

6.4.4 Patch moment quantization

We can embed in maximum $(N-1)$ watermark bits $w_1, w_2, ..., w_{N-1}$ in the obtained N cover patches (the first cover patch $\mathcal{P}_0^w$ is utilized as an anchor patch and thus will not be watermarked as explained in the following). The watermark bit $w_n \in \{0,1\}$ is embedded by quantizing the zero-order moment of the cover patch $\mathcal{P}_n^w$ (the corresponding moment is denoted by $m_{000}^{(\mathcal{P}_n^w)}$). The proposed quantization scheme is a modified version of the conventional scalar Costa scheme (c.f. Chapter 4).

First, a component-wise pseudo-random codebook is established for each moment $m_{000}^{(\mathcal{P}_n^w)}$ as given by Equation (6.9), where $S^{(\mathcal{P}_n^w)}$ is the quantization step, $z \in \mathbb{Z}$ is an integer, $l \in \{0,1\}$ stands for the represented bit of a codeword u, and $t^{(\mathcal{P}_n^w)}S^{(\mathcal{P}_n^w)}$ is the n-th element of an additive pseudo-random dither signal.

$$\mathcal{U}_{m_{000}^{(\mathcal{P}_n^w)},t^{(\mathcal{P}_n^w)}} = \bigcup_{l=0}^{1} \left\{ u = zS^{(\mathcal{P}_n^w)} + l\frac{S^{(\mathcal{P}_n^w)}}{2} + t^{(\mathcal{P}_n^w)}S^{(\mathcal{P}_n^w)} \right\}. \tag{6.9}$$

In our implementation, $t^{(\mathcal{P}_n^w)}, n \in \{1,2,...,N-1\}$ form a simulation sequence of a random variable T uniformly distributed in $\left[-\frac{1}{2}, \frac{1}{2}\right]$ and are generated by inputting a secret key K into an appropriate pseudo-random number generator.

Differently from in the conventional SCS [EBTG03] and from in the wavelet-based watermarking methods presented in Chapter 5, the quantization step $S^{(\mathcal{P}_n^w)}$ in our moment-based scheme is no longer fixed for all the watermarking primitives and is also component-wise. A fixed step, even combined with an adaptive distortion compensation factor value, is experimentally not appropriate for the patch moment quantization. Indeed, different patches can tolerate very different moment variations with respect to the wa-

termark imperceptibility; meanwhile, a same attack can also induce quite different moment variations on different patches. Therefore, in order to ensure a roughly comparable watermarking performance (mainly in terms of imperceptibility and robustness) in different cover patches, we have to set different "adaptive" quantization steps for their zero-order volume moments. We propose the derivation of the component-wise quantization steps $S^{(\mathcal{P}_n^w)}, n \in \{1, 2, ..., N-1\}$ as follows:

$$S^{(\mathcal{P}_n^w)} = \begin{cases} S_{pre}. \left| m_{000}^{(\mathcal{P}_{n-1}^w)'} / \left\lceil \frac{m_{000}^{(\mathcal{P}_{n-1}^w)'}}{m_{000}^{(\mathcal{P}_n^w)}} \right\rceil \right|, & \text{if } \left| \frac{m_{000}^{(\mathcal{P}_{n-1}^w)'}}{m_{000}^{(\mathcal{P}_n^w)}} \right| > 1, \\ S_{pre}. \left| m_{000}^{(\mathcal{P}_{n-1}^w)'} . \left\lfloor \frac{m_{000}^{(\mathcal{P}_n^w)}}{m_{000}^{(\mathcal{P}_{n-1}^w)'}} \right\rfloor \right|, & \text{if else,} \end{cases} \tag{6.10}$$

where $m_{000}^{(\mathcal{P}_{n-1}^w)'}$ is the watermarked moment value of the patch $\mathcal{P}_{n-1}^w$ with $m_{000}^{(\mathcal{P}_0^w)'} = m_{000}^{(\mathcal{P}_0^w)}$ (thus the first cover patch is not watermarked), and S_{pre} is given by:

$$S_{pre} = \begin{cases} 0.04, & \text{if } \left| m_{000}^{(\mathcal{P}_n^w)} \right| > 0.01, \\ 0.07, & \text{if } m_{000}^{-} < \left| m_{000}^{(\mathcal{P}_n^w)} \right| \leq 0.01. \end{cases} \tag{6.11}$$

It can be noticed that the quantization step of the patch $\mathcal{P}_n^w$ is related to the quantized moment of its previous patch $\mathcal{P}_{n-1}^w$. This is partially inspired from the work of Pérez-González et al. [PGMBA05]. Their rational dither modulation (RDH) method achieves the invariance to the value-metric scaling attacks for the quantization index modulation watermarking paradigm [CW01a]. We have proposed the above RDH-like scheme in part to reinforce the watermark robustness against the alteration of the farthest vertex (from the mesh center) that is used by the uniform scaling operation during the mesh normalization step (c.f. Section 6.4.1). This alteration is possible after the watermark embedding or the attacks. The introduction of the term $m_{000}^{(\mathcal{P}_{n-1}^w)'}$ in the calculation of $S^{(\mathcal{P}_n^w)}$ makes the quantization scheme intrinsically invariant to uniform scaling, and thus can effectively enhance the watermark robustness against the local scaling phenomenon caused by the alteration of the farthest vertex.

The quantization step $S^{(\mathcal{P}_n^w)}$ calculated using Equations (6.10) and (6.11) is approximately proportional to the moment amplitude $\left| m_{000}^{(\mathcal{P}_n^w)} \right|$ of the patch to be watermarked. In consequence, the patches with larger moment amplitudes can adaptively have larger moment variations. This is quite reasonable because we can easily prove that a same additive vertex coordinates modification induces larger moment variation on a patch with a larger moment amplitude. There are different S_{pre} values for the patches having moderate moment amplitudes and those having large amplitudes (c.f. Equation (6.11)).

This distinction helps to balance the induced distortions on these different patches and is also theoretically reasonable (please refer to Proof 1 at the end of this chapter). Although a more sophisticated derivation of S_{pre} may be possible, the empirical setting as given by Equation (6.11) has already worked well enough in practice for most meshes.

We can easily find in the constructed codebook $\mathcal{U}_{m_{000}^{(\mathcal{P}_n^w)},t^{(\mathcal{P}_n^w)}}$ the nearest codeword to $m_{000}^{(\mathcal{P}_n^w)}$ that correctly represents the watermark bit w_n. We denote this found codeword by $u_{m_{000}^{(\mathcal{P}_n^w)}}$. The quantized value $m_{000}^{(\mathcal{P}_n^w)'}$ is calculated according to Equation (6.12), where $\alpha^{(\mathcal{P}_n^w)}$ is the DC factor. We always take an appropriate value for $\alpha^{(\mathcal{P}_n^w)}$ so as to ensure the correctness of the watermark extraction when there is no attack. Meanwhile, although our quantization scheme is slightly different from the conventional SCS, we still follow the results of Pérez-Freire et al [PFCPG05], trying to keep $\alpha^{(\mathcal{P}_n^w)}$ close to 0.50 so as to have an *a priori* good watermarking security level.

$$m_{000}^{(\mathcal{P}_n^w)'} = m_{000}^{(\mathcal{P}_n^w)} + \alpha^{(\mathcal{P}_n^w)} \left(u_{m_{000}^{(\mathcal{P}_n^w)}} - m_{000}^{(\mathcal{P}_n^w)} \right). \tag{6.12}$$

It is possible that, after the quantization of the moment of patch $\mathcal{P}_n^w$, the ceiled ($\lceil\ \rceil$) or floored ($\lfloor\ \rfloor$) integer moment ratio between $m_{000}^{(\mathcal{P}_{n-1}^w)'}$ and $m_{000}^{(\mathcal{P}_n^w)'}$ (c.f. Equation (6.10)) may be different from that between $m_{000}^{(\mathcal{P}_{n-1}^w)'}$ and $m_{000}^{(\mathcal{P}_n^w)}$, so that the quantization step $S^{(\mathcal{P}_n^w)}$ can be different at extraction. When this causality problem occurs (in fact it rarely occurs), we re-quantize the corresponding patch volume moment. More precisely, we try to find a quantized value $m_{000}^{(\mathcal{P}_n^w)'}$ in the decoding area of $u_{m_{000}^{(\mathcal{P}_n^w)}}$ that satisfies two constraints: 1) the integer moment ratio mentioned above should be kept unchanged after quantization, with a sufficient margin to the critical value for the continuous ratio between $m_{000}^{(\mathcal{P}_{n-1}^w)'}$ and $m_{000}^{(\mathcal{P}_n^w)'}$; and 2) a minimum robustness should be attained. The robustness is roughly measured by the distance between the quantized value $m_{000}^{(\mathcal{P}_n^w)'}$ and the codeword $u_{m_{000}^{(\mathcal{P}_n^w)}}$. Sometimes it is impossible to find a qualified quantized value which satisfies both constraints. In this case, we have to set $u_{m_{000}^{(\mathcal{P}_n^w)}}$ as the nearest codeword to $m_{000}^{(\mathcal{P}_n^w)}$ that lies in the correct interval with regard to the integer moment ratio and meanwhile correctly represents the watermark bit w_n. We can always find a qualified quantized value in the decoding area of this new codeword, but the induced distortion is normally increased.

6.4.5 Patch deformation

The next step is to deform the cover patches so as to reach their quantized moment values $m_{000}^{(\mathcal{P}_n^w)'}, n \in \{1, 2, ..., N-1\}$. Since it is not easy to deduce the new coordinates of the patch vertices directly from the quantized moment, we need to modify the moment of a cover patch heuristically and iteratively by moving its comprised vertices.

The amplitude and direction of this patch deformation is iteratively adjusted in each step so that the patch zero-order moment gradually achieves its target value. Besides, the displacements of all the vertices within a patch are modulated by using a smooth deformation pattern function that is illustrated in Figure 6.5, so that the patch's global deformation is of low frequency. Each vertex has its own multiplicative deformation factor derived from this mask function and the relative position of the vertex within its belonging patch. For a vertex v_k within $\mathcal{P}_n^w$, the derivation of its deformation factor begins with a normalization of its coordinates:

$$h_k^{'} = 1 - \left| \frac{2(h_k - h_{min}^{(\mathcal{P}_n^w)})}{h_{max}^{(\mathcal{P}_n^w)} - h_{min}^{(\mathcal{P}_n^w)}} - 1 \right| \in [0,1], \tag{6.13}$$

$$\theta_k^{'} = 1 - \left| \frac{2(\theta_k - \theta_{min}^{(\mathcal{P}_n^w)})}{\theta_{max}^{(\mathcal{P}_n^w)} - \theta_{min}^{(\mathcal{P}_n^w)}} - 1 \right| \in [0,1], \tag{6.14}$$

where $h_k^{'}$ and $\theta_k^{'}$ are the normalized coordinates, $h_{max}^{(\mathcal{P}_n^w)}$ and $h_{min}^{(\mathcal{P}_n^w)}$ ($\theta_{max}^{(\mathcal{P}_n^w)}$ and $\theta_{min}^{(\mathcal{P}_n^w)}$) are respectively the maximum and the minimum h domain (θ domain) coordinates of all the vertices within $\mathcal{P}_n^w$ or on the borders of $\mathcal{P}_n^w$. Under this normalization, the vertices close to the patch borders will have small $h_k^{'}$ and $\theta_k^{'}$ values, while the vertices near the patch center will receive large values. For each vertex, two weights are then calculated: Equation (6.15) gives the formula for the h domain weight calculation, the calculation of the θ domain weight $wt_{\theta_k^{'}}$ has a similar form.

$$wt_{h_k^{'}} = \begin{cases} 0 & \text{if } 0 \leq h_k^{'} < 0.1, \\ \frac{1}{2}\sqrt{|s-1|}\left[\sin\left(\frac{5\pi}{3}\left(h_k^{'} - \frac{2}{5}\right)\right) + 1\right] & \text{if } 0.1 \leq h_k^{'} < 0.7, \\ \sqrt{|s-1|} & \text{if } 0.7 \leq h_k^{'} \leq 1.0, \end{cases} \tag{6.15}$$

where s is called the global deformation factor. The individual deformation factor s_{v_k} for vertex v_k is then determined as follows:

$$s_{v_k} = \begin{cases} 1 + wt_{h_k^{'}}.wt_{\theta_k^{'}} & \text{if } s > 1, \\ 1 - wt_{h_k^{'}}.wt_{\theta_k^{'}} & \text{if } s < 1. \end{cases} \tag{6.16}$$

Finally, the coordinates of a candidate displaced vertex are obtained as the multiplication of its original coordinates with s_{v_k} or $(2 - s_{v_k})$, depending on the contribution sign of its incident facets (more details in Algorithm 6.1).

The defined deformation mask (c.f. Figure 6.5) is very smooth: it is constant in the border and center regions, and has a sinus-like shape between the above two regions. Its

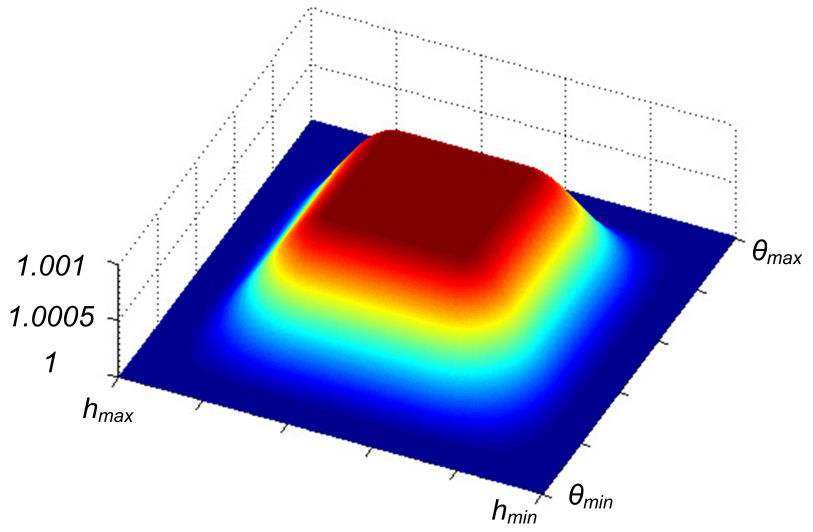


Figure 6.5: Illustration of the deformation mask function pattern (in continuous setting). Here the global deformation factor s is equal to 1.001.

amplitude and direction depend on the global deformation factor s. The objective now is to find, for each patch, the correct value for s that produces the target quantized moment value when applying the deformation mask on the original patch. For this purpose, we have derived a simple and efficient iterative process, which is summarized as Algorithm 6.1. Note that some vertices are not modifiable during the patch deformation. These non-modifiable vertices include the added border vertices, the 1-ring neighbors of the border vertices, and the vertices having simultaneously facets with positive and negative moment contributions. We have also constrained that a displaced vertex cannot get out of its original patch. By using this iterative algorithm, normally the target moment value can be attained within less than 25 iterations. Actually, each patch may have its own deformation mask function (e.g. the ranges of $h_k^{'}$ in Equation (6.15) can be different for different patches); however, a uniform setting of the above mask for all the patches is already satisfactory in practice.

Figure 6.6 illustrates the distortion effects of a moderate-intensity watermark and a very strong-intensity watermark. There exists hardly any visual distortion for the former because the modification is of low frequency; for the latter, the distortion becomes visible and has a similar shape as the deformation mask. In practice, we never use a strong embedding strength as illustrated in Figure 6.6.(c) to watermark a 3-D mesh, in fact a moderate strength as in Figure 6.6.(b) already ensures a very satisfactory robustness.

Notations:

s is the global deformation factor

k_s is the modification step of s

$m_{000}^{(\mathcal{P}_n^w)}$ is the original moment of the patch

$m_{000}^{(\mathcal{P}_n^w)'}$ is the target moment value

m_i is the zero-order moment of the deformed patch obtained after i-th iteration

Determine the comprised vertices for the current patch $\mathcal{P}_n^w$; for each comprised vertex deduce its modifiability; for each modifiable vertex v_k record its original coordinates (x_k, y_k, z_k)

Initialize the parameters: $s = 1$, $k_s = 0.01$, $i = 1$, $\mathrm{m}_{-1} = \mathrm{m}_0 = m_{000}^{(\mathcal{P}_n^w)}$

repeat

Modify s according to the following rule:

- if $\mathrm{m}_{i-1} < m_{000}^{(\mathcal{P}_n^w)'}$ and $\mathrm{m}_{i-2} < m_{000}^{(\mathcal{P}_n^w)'}$, then $s \leftarrow s + k_s$
- if $\mathrm{m}_{i-1} < m_{000}^{(\mathcal{P}_n^w)'}$ and $\mathrm{m}_{i-2} > m_{000}^{(\mathcal{P}_n^w)'}$, then $k_s \leftarrow k_s/2$ and $s \leftarrow s + k_s$
- if $\mathrm{m}_{i-1} > m_{000}^{(\mathcal{P}_n^w)'}$ and $\mathrm{m}_{i-2} > m_{000}^{(\mathcal{P}_n^w)'}$, then $s \leftarrow s - k_s$
- if $\mathrm{m}_{i-1} > m_{000}^{(\mathcal{P}_n^w)'}$ and $\mathrm{m}_{i-2} < m_{000}^{(\mathcal{P}_n^w)'}$, then $k_s \leftarrow k_s/2$ and $s \leftarrow s - k_s$

for each modifiable vertex v_k in $\mathcal{P}_n^w$ **do**

Derive its deformation factor s_{v_k} (Equations (6.15) and (6.16)) according to s and its normalized coordinates (Equations (6.13) and (6.14))

Modify its original coordinates to obtain a candidate displaced vertex v_k' by using the following rule:

- if all incident facets of v_k have positive moment contributions, then $(x_k', y_k', z_k') = s_{v_k}.(x_k, y_k, z_k)$
- if all incident facets of v_k have negative moment contributions, then $(x_k', y_k', z_k') = (2 - s_{v_k}).(x_k, y_k, z_k)$

end for

evaluate m_i as the zero-order moment of the obtained deformed patch

iteration number incrementation: $i = i + 1$

until $\left|\mathrm{m}_i - m_{000}^{(\mathcal{P}_n^w)'}\right| < \epsilon$ or $i = I_{max}$

Algorithm 6.1: Iterative patch deformation algorithm.

Figure 6.6: This figure illustrates the visual effects of the patch deformation: (a) the original Venus, (b) a moderately watermarked Venus, and (c) a strongly watermarked Venus.

6.4.6 Moment compensation

The objective of this step is to recover the original center position and principal axis orientations of the cover mesh. Concretely, we need to compensate for the variations of the mesh's m_{100}, m_{010}, m_{001}, m_{110}, m_{101} and m_{011} moments that have been induced by the cover patch deformations in the last step, so that these six moments all become zero again, or at least reasonably small.

Our moment compensation method is based on the following property of the iterative patch deformation process described in Algorithm 6.1: when deforming a patch by using this algorithm, it can be proven (please refer to Proof 2 at the end of this chapter) and has also been experimentally validated that the moment variation ratios $\frac{\Delta m_{100}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$, $\frac{\Delta m_{010}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$, $\frac{\Delta m_{001}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$, $\frac{\Delta m_{110}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$, $\frac{\Delta m_{101}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$ and $\frac{\Delta m_{011}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$ are kept approximately constant under different values of the global deformation factor s. The compensation patches $\mathcal{P}_l^c, l \in \{0, 1, .., 11\}$ are deformed arbitrarily by using Algorithm 6.1 prior to the moment compensation step so as to learn the values of these ratios (of course these 12 patches are then restored to their initial position). For the sake of notation simplicity, the six learned ratios of the compensation patch $\mathcal{P}_l^c$ are hereafter denoted by r_1^l to r_6^l.

The problem is then formulated as the deduction of the correct moment variations Δm_{000}^l for the 12 compensation patches such that the variations of the other moments compensate for the global moments $\tilde{m}_{100}$, $\tilde{m}_{010}$, $\tilde{m}_{001}$, $\tilde{m}_{110}$, $\tilde{m}_{101}$ and $\tilde{m}_{011}$ of the obtained mesh after the watermark bit embedding through cover patch deformation. A 6×12 linear least-squares system is constructed:

$$\check{M} = \arg\min_{M} \left\| R.M - \tilde{M} \right\|_2^2, \tag{6.17}$$

where R is a 6×12 matrix with $R_{ij} = \mathrm{r}_i^{j-1}$, M is a 12×1 matrix with $M_{i1} = \Delta m_{000}^{i-1}$, and $\tilde{M} = [\tilde{m}_{100} \;\; \tilde{m}_{010} \;\; \tilde{m}_{001} \;\; \tilde{m}_{110} \;\; \tilde{m}_{101} \;\; \tilde{m}_{011}]^T$. The optimization of the above system is subject to two constraints:

$$Lb \leq M \leq Ub, \tag{6.18}$$

$$R^{'}.M^{'} = \tilde{M}^{'}, \tag{6.19}$$

where Lb and Ub represent respectively the lower and upper bounds of the moment variations, and $R^{'}$, $M^{'}$ and $\tilde{M}^{'}$ are composed of the last three rows of R, M and $\tilde{M}$, respectively. The first constraint is related to the deformation imperceptibility. Indeed, we have selected some appropriate values for the moment variation lower and upper bounds so that the deformation amplitude of the compensation patches is of the same order as that of the cover patches. The second constraint defines the priority of compensating the second-order moments. The introduction of this second constraint is based on the observation that our whole watermarking algorithm is experimentally much more sensitive to the principal axis orientation change than to the mesh center change.

Then, we solve the least-squares system in Equation (6.17) subject to the above two constraints and deduce the moment variations (and thus the target zero-order moment values) for the 12 compensation patches. These patches are afterward deformed by using Algorithm 6.1 so as to attain the wanted moment values. After this step, the six compensated first and second order moments of the obtained mesh are very close to zero and do not have any negative influence on the blind watermark extraction.

The last step of the watermark embedding procedure is the removal of the auxiliary vertices and edges that were inserted during the patch decomposition step. Finally, we obtain a stego mesh with a multi-bit watermark embedded in it.

6.5 Watermark Extraction

The watermark extraction algorithm is blind and computationally efficient. First, the input mesh is normalized by using the technique described in Section 6.4.1. Then, the vertex coordinates are converted into cylindric system and the mesh is decomposed into patches by discretizing its h and θ domains. After using the patch classification rules presented in Section 6.4.3, we can pick out the candidate cover patches for the watermark bit extraction. Next, with the knowledge of the secret key K and by using Equations (6.9) to (6.11), we construct a codebook $\mathcal{U}_{\hat{m}_{000}^{(\mathcal{P}_n^w)}, t^{(\mathcal{P}_n^w)}}$ for each cover patch $\mathcal{P}_n^w$. According to the

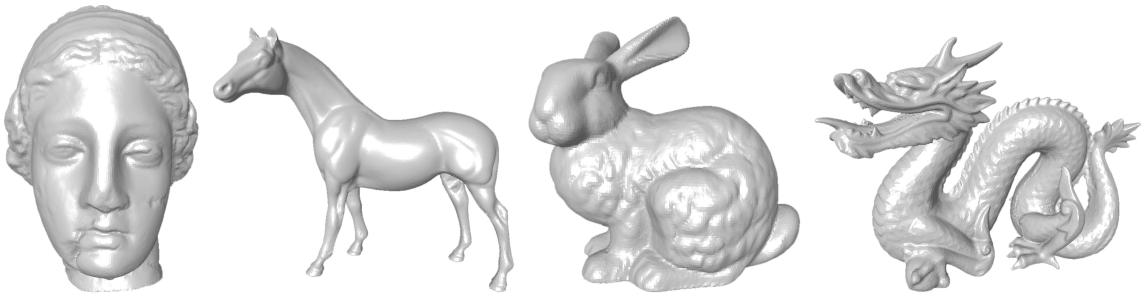


Figure 6.7: The original non-watermarked meshes: (a) Venus, (b) Horse, (c) Bunny, (d) Dragon.

actual moment value $\hat{m}_{000}^{(\mathcal{P}_n^w)}$ of the patch, we can find the codeword $u_{\hat{m}_{000}^{(\mathcal{P}_n^w)}}$ that is the closest to $\hat{m}_{000}^{(\mathcal{P}_n^w)}$ in the constructed codebook. Finally, the n-th extracted watermark bit $w_n^{'}$ is considered as the represented bit of the codeword $u_{\hat{m}_{000}^{(\mathcal{P}_n^w)}}$.

6.6 Experimental Results

6.6.1 Basic simulations

The proposed watermarking method has been tested on several meshes. Figure 6.7 illustrates four of them: Venus (100759 vertices), Horse (112642 vertices), Bunny (34835 vertices) and Dragon (50000 vertices). The adjustable parameters of our algorithm are the DC factors $\alpha^{(\mathcal{P}_n^w)}$ for the cover patches, which drive the trade-off between distortion, robustness and security. They have been fixed for different meshes (c.f. the second row of Table 6.2) by following two empirical rules: 1) they cannot be too large for the sake of watermarking security, and 2) the meshes having lower vertex sampling density can tolerate larger DC factor values since a stronger deformation can be introduced on them without being noticed. Figure 6.8 illustrates the watermarked meshes. We can hardly observe any visual distortion introduced by the watermark embedding, even on very smooth regions such as the body of the Horse. The main reason is that these induced distortions are smooth and of low frequency, to which the human eyes are not sensitive. Figure 6.9 illustrates the maps of the geometric objective distortions between original and watermarked meshes. It can be noticed that although the distortion is globally well balanced, there still exist some patches that are much more deformed than others. The understanding and improvement of this point constitute one important part of our future work concerning this moment-based watermarking method.

Table 6.2 presents some statistics about the watermark embedding and extraction. All the tests were carried out on a Pentium IV 2.0GHz processor with 2GB memory.

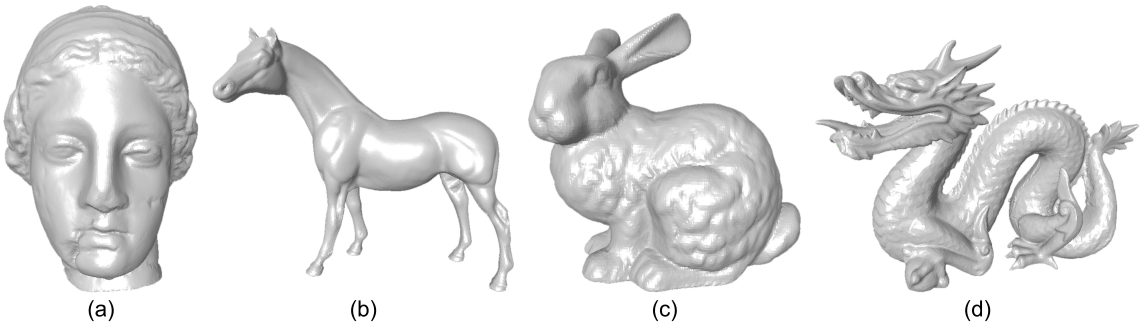


Figure 6.8: The watermarked meshes: (a) Venus, (b) Horse, (c) Bunny, (d) Dragon.

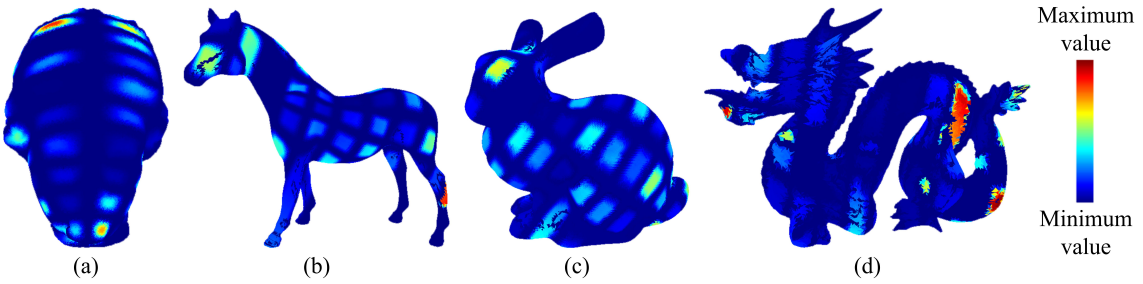


Figure 6.9: The objective distortion maps of the watermarked meshes: (a) Venus, (b) Horse, (c) Bunny, (d) Dragon.

The objective and perceptual distances between watermarked and original meshes are respectively measured by MRMS and MSDM (c.f. Section 5.5.1). One advantage of our method is that it can introduce relatively high-amplitude deformation while keeping it imperceptible. Most of the embedding time is spent on the iterative deformation step, which does not only depend on the mesh size (i.e. its vertex number), but also on its cover patch number. The extraction time is almost completely due to the patch decomposition and is basically proportional to the mesh size.

6.6.2 Robustness against geometry attacks

In the following subsections, the resistance of the embedded watermark is tested under different types of attacks. The robustness is evaluated in terms of the BER (bit error rate)

Table 6.2: Baseline evaluations of the proposed watermarking method.

Mesh model ⇒	Venus	Horse	Bunny	Dragon
$\alpha^{(\mathcal{P}_n^w)}$	0.70	0.75	0.80	0.85
Embedding time (s)	410.8	191.5	109.4	166.2
Extraction time (s)	3.2	2.9	1.1	1.6
WM capacity (bit)	75	46	67	49
MRMS by WM (10^{-3})	2.34	1.04	1.75	1.76
MSDM by WM	0.15	0.17	0.19	0.20

Table 6.3: Robustness against random noise addition.

Model	Amplitude	MRMS (10^{-3})	BER	Correlation
Venus	0.10%	0.33	0.03	0.94
	0.30%	0.98	0.06	0.87
	0.50%	1.63	0.11	0.78
	non-unif. 0.30%	0.68	0.05	0.89
	non-unif. 0.50%	1.13	0.13	0.73
Horse	0.10%	0.21	0.01	0.98
	0.30%	0.64	0.08	0.86
	0.50%	1.07	0.12	0.77
	non-unif. 0.30%	0.45	0.04	0.92
	non-unif. 0.50%	0.78	0.11	0.78
Bunny	0.10%	0.22	0.01	0.98
	0.30%	0.66	0.07	0.85
	0.50%	1.11	0.11	0.77
	non-unif. 0.30%	0.50	0.02	0.95
	non-unif. 0.50%	0.82	0.07	0.85
Dragon	0.10%	0.24	0.01	0.98
	0.30%	0.72	0.12	0.76
	0.50%	1.20	0.19	0.61
	non-unif. 0.30%	0.63	0.14	0.72
	non-unif. 0.50%	0.94	0.24	0.53

of the extracted watermark and the normalized correlation between the extracted and the originally embedded watermark sequences (c.f. Section 5.5.2).

Our watermark is experimentally invariant to all the content preserving attacks. Tables 6.3, 6.4 and 6.5 respectively present the evaluation results of the watermark robustness against noise addition, smoothing and vertex coordinates quantization. In these tables, the attack-induced MRMS distances are also provided. Some geometrically attacked models are illustrated in Figure 6.10.(a)-(d). Our algorithm demonstrates a fairly high robustness against geometry attacks, even those with strong amplitudes or those that are spatially non-uniform (anisotropic). For instance, in average, we can still successfully extract up to 87% of the watermark under 0.50% noise addition (the visual effect of this attack is illustrated in Figure 6.10.(a)). The watermarks embedded in Bunny and Dragon are less robust against smoothing because this attack produces an important shrinking effect on these two models.

6.6.3 Robustness against connectivity attacks

The tested connectivity attacks include surface simplification (spatially uniform and non-uniform), subdivision and remeshing. The used mesh simplification algorithm is Garland and Heckbert's quadric-error-metric-based method [GH97]. The subdivision attacks include the simple midpoint scheme, the modified butterfly scheme and the Loop scheme [ZS00]. The remeshing attack is a uniform resampling of the mesh vertices using the ReMESH software [AF06]; two different target vertex numbers are considered

Table 6.4: Robustness against Laplacian smoothing ($\lambda = 0.03$).

Model	Iteration	MRMS (10^{-3})	BER	Correlation
Venus	10	0.12	0.04	0.92
	50	0.51	0.04	0.92
	100	0.88	0.08	0.84
Horse	10	0.07	0	1
	50	0.29	0.07	0.87
	100	0.52	0.13	0.74
Bunny	10	0.26	0.13	0.73
	30	0.69	0.19	0.62
	50	1.04	0.37	0.27
Dragon	10	0.31	0.08	0.84
	30	0.82	0.24	0.52
	50	1.28	0.41	0.19

Table 6.5: Robustness against uniform vertex coordinates quantization.

Model	Intensity	MRMS (10^{-3})	BER	Correlation
Venus	9-bit	0.66	0.04	0.92
	8-bit	1.32	0.11	0.81
	7-bit	2.70	0.11	0.79
Horse	9-bit	0.49	0	1
	8-bit	0.97	0.15	0.70
	7-bit	2.05	0.26	0.49
Bunny	9-bit	0.52	0.04	0.91
	8-bit	1.05	0.04	0.91
	7-bit	2.07	0.15	0.70
Dragon	9-bit	0.57	0.02	0.96
	8-bit	1.13	0.18	0.63
	7-bit	2.29	0.39	0.23

Table 6.6: Robustness against surface simplification.

Model	Vertex reduction ratio	MRMS (10^{-3})	BER	Correlation
Venus	90%	0.29	0.03	0.95
	95%	0.51	0.05	0.89
	97.5%	0.91	0.07	0.84
	non-unif. 50%	0.25	0.04	0.92
	non-unif. 75%	0.67	0.09	0.82
Horse	90%	0.13	0	1
	95%	0.24	0.02	0.96
	97.5%	0.43	0.07	0.87
	non-unif. 50%	0.21	0.09	0.83
	non-unif. 75%	0.35	0.11	0.78
Bunny	70%	0.21	0	1
	90%	0.54	0.13	0.73
	95%	0.95	0.13	0.74
	non-unif. 25%	0.17	0	1
	non-unif. 50%	0.66	0.13	0.73
Dragon	70%	0.37	0	1
	90%	1.00	0.22	0.56
	95%	1.79	0.46	0.08
	non-unif. 25%	0.23	0	1
	non-unif. 50%	0.86	0.16	0.67

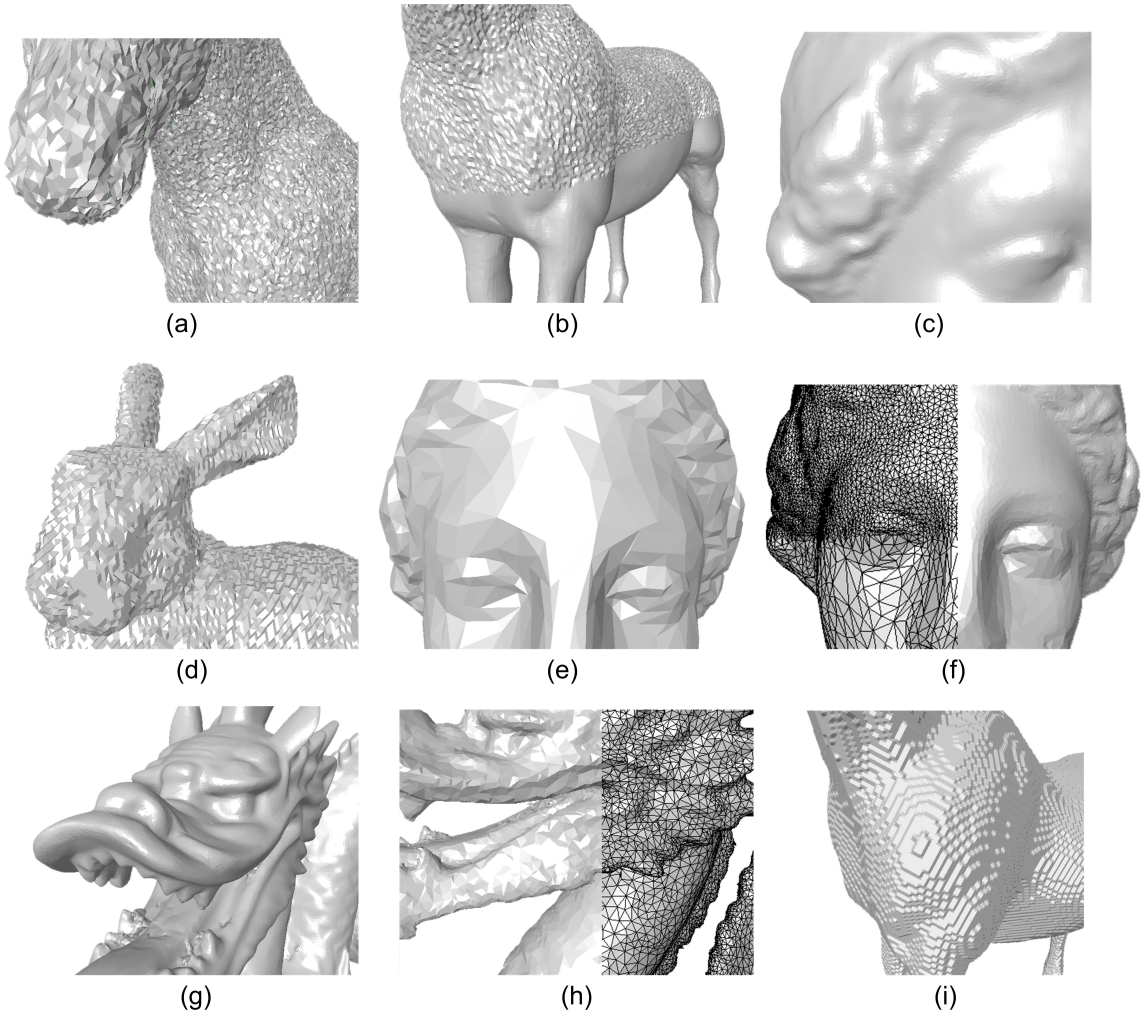


Figure 6.10: Close-ups of some attacked watermarked models: (a) 0.50% random additive noise (BER= 0.12); (b) 0.50% spatially non-uniform noise (BER= 0.11); (c) 100-iteration Laplacian smoothing with $\lambda = 0.03$ (BER= 0.08); (d) 7-bit coordinate quantization (BER= 0.15); (e) spatially uniform simplification by 97.5% vertex reduction (BER= 0.07); (f) spatially non-uniform simplification by 75% vertex reduction, the upper and lower parts are simplified with different reduction ratios (BER= 0.09); (g) 1 Loop subdivision (BER= 0.06); (h) uniform remeshing with original vertex number (BER= 0.10); (i) output mesh of the marching cubes algorithm on a $350 \times 350 \times 350$ discretized (voxelized) Horse (BER= 0.11).

Table 6.7: Robustness against one-step subdivision.

Model	Scheme	MRMS (10^{-3})	BER	Correlation
Venus	Midpoint	0	0.03	0.95
	m-Butterfly	0.10	0.03	0.95
	Loop	0.11	0.04	0.92
Horse	Midpoint	0	0	1
	m-Butterfly	0.05	0	1
	Loop	0.06	0	1
Bunny	Midpoint	0	0	1
	m-Butterfly	0.23	0	1
	Loop	0.23	0.15	0.71
Dragon	Midpoint	0	0	1
	m-Butterfly	0.24	0.02	0.96
	Loop	0.25	0.06	0.88

Table 6.8: Robustness against uniform surface remeshing.

Model	Vertex number	MRMS (10^{-3})	BER	Correlation
Venus	100%	0.08	0.04	0.92
	50%	0.30	0.04	0.92
Horse	100%	0.06	0	1
	50%	0.18	0.04	0.91
Bunny	100%	0.39	0.03	0.94
	50%	0.63	0.13	0.74
Dragon	100%	0.40	0.10	0.80
	50%	1.54	0.45	0.11

for this resampling: they are respectively 100% and 50% of the original vertex number of the watermarked mesh.

Tables 6.6, 6.7 and 6.8 present the obtained robustness evaluation results against the considered connectivity attacks. In Figure 6.10.(e)-(h), some attacked models are illustrated. It can be observed that our scheme possesses a very strong robustness against all these attacks, which are in general considered quite difficult to handle for a blind mesh watermarking algorithm. As an example, for Venus and Horse, we can still retrieve 93% of the watermark bits after having removed 97.5% of the vertices in the models. The watermark embedded in Dragon is less robust against these connectivity attacks since the model has a relatively low number of vertices regarding its complexity, therefore modifying its connectivity induces an important modification on the model's 3-D shape.

6.6.4 Robustness against representation conversion

We have tested one scenario of this serious attack: the watermarked mesh is discretized into a $350 \times 350 \times 350$ voxel grid. In order to extract the watermark from this discrete volumetric representation, we transform it back into a mesh representation by using the well known marching cubes algorithm [LC87]. The watermark extraction is then carried out on this reconstructed mesh. Table 6.9 presents the robustness results under

Table 6.9: Robustness against voxelization.

Model	Resolution	MRMS (10^{-3})	BER	Correlation
Venus	$350 \times 350 \times 350$	0.95	0.13	0.74
Horse	$350 \times 350 \times 350$	1.22	0.11	0.78
Bunny	$350 \times 350 \times 350$	0.85	0.12	0.76
Dragon	$350 \times 350 \times 350$	7.27	0.55	-0.11

this attack. For Venus, Horse and Bunny, the robustness is very satisfactory (the BER is around 0.12), considering the strength of this attack (c.f. Figure 6.10.(i)). The watermark extraction on Dragon fails because the marching cubes algorithm has created very strong artefacts on its tail, which significantly changes the mesh's center and principal axes.

6.6.5 Discussion and comparison

In this subsection, we provide some discussions on the strengths and shortcomings of the proposed watermarking method. We will also compare our method with two recent schemes from Cho et al. [CPJ07], which are deemed to be the most robust blind mesh watermarking algorithms proposed so far. We have applied their algorithms on Horse (Algorithm I) and Bunny (Algorithm II) so as to compare the results in terms of imperceptibility and robustness.

First of all, concerning the watermark imperceptibility, the induced patch deformation in our scheme is of low frequency while their methods seem to introduce relatively high frequency distortions. Figure 6.11 illustrates the Horse and Bunny models watermarked by their and our methods. The objective MRMS distances introduced by their watermark embedding (0.51×10^{-3} for Horse and 0.29×10^{-3} for Bunny) are smaller, but these small-amplitude objective distortions seem to be more perceptible (c.f. Figure 6.11.(a) and (c)). This point is also confirmed by the MSDM perceptual distances between their watermarked and original models (0.23 for Horse and 0.32 for Bunny against respectively 0.17 and 0.19 for models watermarked by our method). In particular, some ring-like high frequency artefacts may occur on the surface of their watermarked meshes, especially on smooth regions such as the body of the Horse. Indeed, in their methods, the watermark is embedded through modification of the vertex norm histogram, while not considering the relative positions of the involved vertices. Contrarily, during our watermark embedding process, the deformation pattern of each patch is carefully controlled by modulating the vertex displacements with a smooth mask function. Besides, it can be observed from Tables 6.3-6.9 that our watermark embedding may induce a similar MRMS distance as some strong-amplitude attacks. This illustrates the effectiveness of our mask-based patch deformation algorithm as well as the necessity of performing

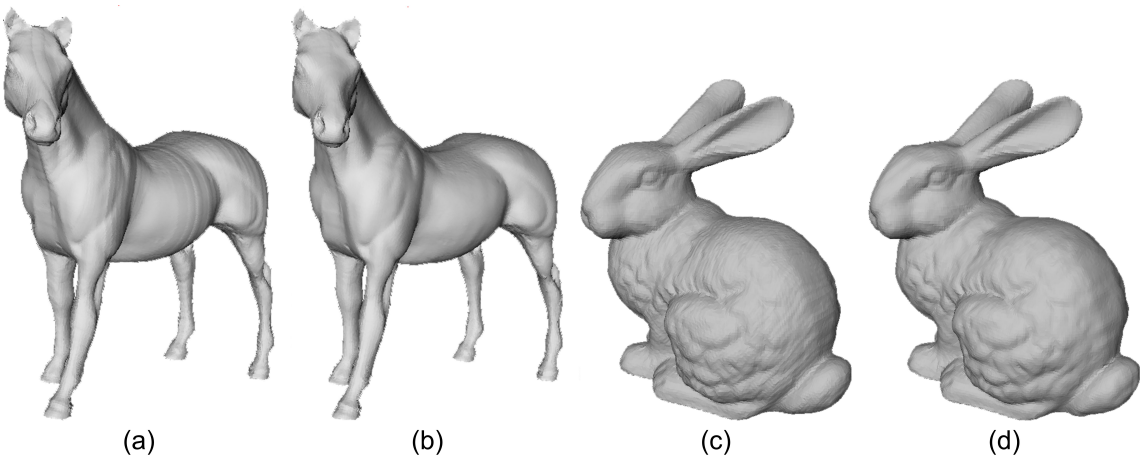


Figure 6.11: Imperceptibility comparison between the algorithms of Cho et al. and our method: (a) Horse watermarked by their Algorithm I (strength parameter $\alpha = 0.03$); (b) Horse watermarked by our method; (c) Bunny watermarked by their Algorithm II (strength parameter $\alpha = 0.07$); (d) Bunny watermarked by our method.

the moment compensation post-processing.

Tables 6.10 and 6.11 present the robustness evaluation results of the watermarks embedded in the stego models of Cho et al. that are illustrated in Figure 6.11.(a) and (c). The corresponding correlation values of our method are also listed in the last column of these tables. This robustness comparison is under the premise of a same capacity on a same mesh model for both kinds of methods. Our stego Horse that has nearly no visual distortions is much more robust (under both geometry and connectivity attacks) than their stego Horse on which there exist noticeable distortions. Our algorithm is particularly more robust against quantization (our correlation is 1 against 0.66 for their method, under a 9-bit quantization) and simplification (1 against 0.58 for correlation values under a 90% simplification). Our stego Bunny has also a better imperceptibility than their stego Bunny, and is more robust against connectivity attacks (especially under simplification). Robustness against geometry attacks is quite similar for the two stego Bunny models: our algorithm is globally more robust to strong distortions while their method performs better against the attacks with small amplitudes. One exception is the smoothing attack which introduces obvious shrinkage deformations on this relatively sparse surface and thus leads to a bad performance of our method. In general, their methods have difficulties under strong-amplitude non-uniform simplifications since the calculated mesh center can be mistakenly moved toward the mesh part where the vertex density is higher. In all, our method is particularly suitable for the protection of dense meshes, for which the imperceptibility and the robustness against simplification are the main concerns. The advantage of their algorithms is that the watermark can resist

Table 6.10: Robustness evaluation results on the watermarked Horse by Algorithm I of Cho et al. ($\alpha = 0.03$, 46 bits are embedded)

Attack	BER	Correlation	Our Corr.
0.10% noise	0	1	0.98
0.30% noise	0.24	0.52	0.86
0.50% noise	0.41	0.17	0.77
10-itera. smoothing	0	1	1
50-itera. smoothing	0.09	0.84	0.87
100-itera. smoothing	0.20	0.62	0.74
9-bit quantization	0.17	0.66	1
8-bit quantization	0.37	0.26	0.70
7-bit quantization	0.46	0.08	0.49
90% simplification	0.22	0.58	1
95% simplification	0.22	0.57	0.96
97.5% simplification	0.30	0.40	0.87
50% non-unif. simplifi.	0.11	0.80	0.83
75% non-unif. simplifi.	0.22	0.56	0.78
100% uniform remeshing	0	1	1
50% uniform remeshing	0.24	0.52	0.91

attacks that introduce much higher objective distortions than its embedding. Neither method achieves the robustness against strong local deformation and cropping.

One drawback of our watermarking scheme is that its capacity depends on the mesh shape and normally varies from 45 bits to 75 bits. On the contrary, the capacity of the methods of Cho et al. is independent from the specific shape of the cover mesh and thus can ensure a constant capacity, say 64 bits, for all the 3-D models. In the future, we would like to exploit the possibility of ensuring a minimum capacity for our method while keeping the other performances. Finally, our method outperforms their algorithms in terms of security. In the latter, no secret key is used and the modified histogram is exposed to everyone, including the pirates. It seems that the intrinsic easy accessibility of the global vertex norm histogram makes the security improvement difficult. On the contrary, a secret key is used in our algorithm for the SCS-like moment quantization and the current parameter setting seems to ensure a relatively good secrecy of this key.

6.7 Conclusion

In this chapter, a new robust and blind polygonal mesh watermarking algorithm is proposed. The watermark bits are embedded by slightly deforming some selected cover patches obtained after a simple mesh decomposition in the cylindric coordinate system. Watermark imperceptibility is ensured by using a smooth low-frequency mask to modulate the patch deformation; besides, the causality problem is solved by introducing a compensation post-processing step. The robustness of this approach is due to the stability of the global and local (in patches) volume moment values under geometry,

Table 6.11: Robustness evaluation results on the watermarked Bunny by Algorithm II of Cho et al. ($\alpha = 0.07$, 64 bits are embedded)

Attack	BER	Correlation	Our Corr.
0.10% noise	0	1	0.98
0.30% noise	0	1	0.85
0.50% noise	0.17	0.69	0.77
10-itera. smoothing	0.03	0.94	0.73
30-itera. smoothing	0.16	0.69	0.62
50-itera. smoothing	0.22	0.57	0.27
9-bit quantization	0.02	0.97	0.91
8-bit quantization	0.06	0.88	0.91
7-bit quantization	0.47	0.07	0.70
70% simplification	0.09	0.81	1
90% simplification	0.34	0.32	0.73
95% simplification	0.55	−0.09	0.74
25% non-unif. simplifi.	0.07	0.87	1
50% non-unif. simplifi.	0.48	0.03	0.73
midpoint subdivision	0.02	0.97	1
m-butterfly subdivision	0.02	0.97	1
Loop subdivision	0.09	0.81	0.71
100% uniform remeshing	0.02	0.97	0.94
50% uniform remeshing	0.22	0.57	0.74

connectivity and even representation conversion attacks as long as they do not seriously modify the intrinsic shape (i.e. visual appearance) of the mesh.

Proof 1

In this proof, we will demonstrate that it is theoretically reasonable for patches with high m_{000} moment amplitudes to receive a small S_{pre} value.

Recall that S_{pre} is involved in the determination of the component-wise quantization step $S^{(\mathcal{P}_n^w)}$ in Equations (6.10) and (6.11) in Section 6.4.4.

For the sake of simplicity, we consider a simple patch composed of only one triangle facet $f = \{v_1, v_2, v_3\} = \{(x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3)\}$ and its uniformly scaled version $f' = \left\{v_1', v_2', v_3'\right\} = \{(k.x_1, k.y_1, k.z_1), (k.x_2, k.y_2, k.z_2), (k.x_3, k.y_3, k.z_3)\}$ ($k > 1$). We then make the assumption that the moment quantization on these two patches should introduce comparable variations on the coordinates of their comprised vertices, in order to ensure a uniform deformation. Note that here we take into account only the objective distance metric without perceptual considerations, still for the sake of simplicity.

Now, assume that f and f' have positive zero-order moments and are subject to a same facet vertex coordinate variation $(\Delta x_1, \Delta y_1, \Delta z_1), (\Delta x_2, \Delta y_2, \Delta z_2), (\Delta x_3, \Delta y_3, \Delta z_3)$, which simulates the consequence of the watermark embedding. After neglecting the second and higher order terms (e.g. $\Delta x_1.\Delta y_2.z_3$ and $\Delta x_1.\Delta y_2.\Delta z_3$) in the moment cal-

culation formulas, we can easily find out that the following relationship approximately holds:

$$\Delta m_{000}^{(f')} = k^2 \Delta m_{000}^{(f)}, \tag{6.20}$$

where $\Delta m_{000}^{(f')}$ and $\Delta m_{000}^{(f)}$ are respectively the moment variations of f' and f. Considering that $m_{000}^{(f')} = k^3 m_{000}^{(f)}$, we then obtain

$$\frac{\Delta m_{000}^{(f')}}{m_{000}^{(f')}} = \frac{1}{k} . \frac{\Delta m_{000}^{(f)}}{m_{000}^{(f)}}. \tag{6.21}$$

The reason for neglecting the second and higher order terms is explained as follows. Indeed, the vertex coordinate alteration during the watermark embedding is quite small (in the order of 0.10%). Therefore, the second-order terms are much smaller than the first-order terms (also of about 0.10%). Hence, these small-value terms can be neglected in the above analysis without introducing significant errors.

From Equation (6.10) which presents the calculation of the component-wise quantization step $S^{(\mathcal{P}_n^w)}$, we can see that the terms $\left| m_{000}^{(\mathcal{P}_{n-1}^w)'} / \left\lceil \frac{m_{000}^{(\mathcal{P}_{n-1}^w)'}}{m_{000}^{(\mathcal{P}_n^w)}} \right\rceil \right|$ and $\left| m_{000}^{(\mathcal{P}_{n-1}^w)'} . \left\lfloor \frac{m_{000}^{(\mathcal{P}_n^w)}}{m_{000}^{(\mathcal{P}_{n-1}^w)'}} \right\rfloor \right|$ are approximately equal to the moment amplitude of the current patch (i.e. $m_{000}^{(f)}$ or $m_{000}^{(f')}$ in Equation (6.21)). Meanwhile, the final moment variation (i.e. $\Delta m_{000}^{(f)}$ or $\Delta m_{000}^{(f')}$ in Equation (6.21)) is also somewhat proportional to the quantization step $S^{(\mathcal{P}_n^w)}$. Therefore (c.f. Equation (6.10)), the term S_{pre} approximately represents the ratio between the moment variation and the original moment value of the current patch. From Equation (6.21), we can deduce that, in order to have comparable vertex variations for the two patches under consideration, the above mentioned ratio should be smaller for the patch having a higher m_{000} value due to the existence of the term $\frac{1}{k}$ $(k > 1)$ on the right side of the formula.

In all, although the above deduction makes many assumptions and considers only a very simple case, it potentially constitutes a reasonable proof for setting a small S_{pre} value for patches with high m_{000} moment amplitudes. This measure has also been demonstrated effective in practice since it can, to some extent, balance the final watermark induced distortions in different patches with different sizes.

□

Proof 2

In this proof, we will demonstrate the following property: when deforming a patch $\mathcal{P}_j$ by using Algorithm 6.1, the moment variation ratios $\frac{\Delta m_{100}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$, $\frac{\Delta m_{010}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$, $\frac{\Delta m_{001}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$, $\frac{\Delta m_{110}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$, $\frac{\Delta m_{101}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$

and $\frac{\Delta m_{011}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$ are kept approximately constant under different s (i.e. global deformation factor) values.

We will take $\frac{\Delta m_{100}^{(\mathcal{P}_j)}}{\Delta m_{000}^{(\mathcal{P}_j)}}$ as an example to carry out the demonstration and hereafter neglect the patch designation superscripts in the notations of the volume moments and their variations.

First, we can rewrite the m_{000} calculation formula for a patch composed of vertices $v_1, v_2, ..., v_N$ as the following sum of several multiplication terms of three vertex coordinates:

$$m_{000} = \frac{1}{6} \sum_{i,j,k} \text{sign}_{ijk}.x_i y_j z_k, \tag{6.22}$$

where $\text{sign}_{ijk} \in \{-1, 1\}$ and the triplet $x_i y_j z_k$ only occurs if v_i, v_j and v_k are within a same facet.

For the sake of simplicity, we suppose hereafter that all the facets in the patch have positive moment contributions and $s > 1$. Under the proposed modulated patch deformation, x_i becomes $s_{v_i}.x_i$ after a displacement, where $s_{v_i} = 1 + wt_{h_i'}.wt_{\theta_i'}$. It is easy to deduce (c.f. Equation (6.15)) that the above two weights can be rewritten as:

$$wt_{h_i'} = a_1.\sqrt{(s-1)}, \tag{6.23}$$

$$wt_{\theta_i'} = a_2.\sqrt{(s-1)}. \tag{6.24}$$

Note that a_1 and a_2 only depend on the normalized coordinates of the vertex v_i and the shape of the modulation function, which are invariant under different s values. Thus, we obtain $s_{v_i} = 1 + a_1 a_2 (s-1)$ and $x_i^{'} = x_i + a_1 a_2 (s-1) x_i = x_i + a_i (s-1) x_i$, with $a_i = a_1 a_2$. After neglecting the second and higher order terms in the moment calculation formulas, we can deduce the m_{000} moment variation as follows:

$$\begin{aligned} \Delta m_{000} &= \frac{1}{6} \sum_{i,j,k} \left(a_i (s-1) x_i y_j z_k + a_j (s-1) x_i y_j z_k + a_k (s-1) x_i y_j z_k \right) \\ &= \frac{1}{6} (s-1) \sum_{i,j,k} \left(a_i x_i y_j z_k + a_j x_i y_j z_k + a_k x_i y_j z_k \right). \end{aligned} \tag{6.25}$$

Similarly, we can deduce the m_{100} moment variation approximately as:

$$\begin{aligned} \Delta m_{100} &= \frac{1}{24} \sum_{i,j,k,l} (a_i (s-1) x_i x_j y_k z_l + a_j (s-1) x_i x_j y_k z_l + a_k (s-1) x_i x_j y_k z_l \\ &\quad + a_l (s-1) x_i x_j y_k z_l) \\ &= \frac{1}{24} (s-1) \sum_{i,j,k,l} \left(a_i x_i x_j y_k z_l + a_j x_i x_j y_k z_l + a_k x_i x_j y_k z_l + a_l x_i x_j y_k z_l \right). \end{aligned} \tag{6.26}$$

Hence, the ratio between Δm_{100} and Δm_{000} is equal to:

$$\begin{aligned}\frac{\Delta m_{100}}{\Delta m_{000}} &= \frac{\frac{1}{24}(s-1)\sum_{i,j,k,l}\left(a_i x_i x_j y_k z_l + a_j x_i x_j y_k z_l + a_k x_i x_j y_k z_l + a_l x_i x_j y_k z_l\right)}{\frac{1}{6}(s-1)\sum_{i,j,k}\left(a_i x_i y_j z_k + a_j x_i y_j z_k + a_k x_i y_j z_k\right)} \\ &= \frac{1}{4}.\frac{\sum_{i,j,k,l}\left(a_i x_i x_j y_k z_l + a_j x_i x_j y_k z_l + a_k x_i x_j y_k z_l + a_l x_i x_j y_k z_l\right)}{\sum_{i,j,k}\left(a_i x_i y_j z_k + a_j x_i y_j z_k + a_k x_i y_j z_k\right)}. \end{aligned} \tag{6.27}$$

Finally, we can conclude that the above moment variation ratio is completely determined by the original coordinates of the patch vertices (under a fixed modulation function shape) and thus is independent of the global deformation factor s.

□

Chapter 7

Robust and Blind Mesh Watermarking Based on Manifold Harmonics Transform

Contents

In this chapter, we present a blind and robust 3-D mesh watermarking scheme that makes use of the recently proposed manifold harmonics transform. The mesh spectrum coefficient amplitudes obtained by using this transform are quite robust against various attacks, including connectivity changes. A blind multi-bit watermark is embedded through an iterative SCS quantization of the low frequency coefficient amplitudes. The main strength of the proposed watermarking scheme is its very high imperceptibility due to the fact that the human visual system is insensitive to the modifications of the mesh low frequency components. Our watermark is experimentally robust against both geometry and connectivity attacks. Comparison results with two state-of-the-art

algorithms are provided.

This research on spectral mesh watermarking is a joint work between the LIRIS Laboratory and the Department of Computer Science of the University of York in England. This work has led to a paper which has been accepted by an international conference [WLBD09].

7.1 Introduction and Motivation

In general, spectral mesh watermarking methods have the advantage of being more imperceptible. Indeed, the human visual system is less sensitive to the modifications of the low and medium frequency components of a 3-D mesh. Accordingly, the watermarks embedded in these components are demonstrated to be more invisible. Furthermore, after a spectral-to-spatial inverse transformation, the modifications in the spectral domain due to the watermark embedding will spread to all the spatial parts of the mesh; therefore, it is less likely for a spectral mesh watermarking method to introduce noticeable distortions (e.g. with specific spatial patterns) on the mesh surface. In a general sense, this also helps to enhance the watermarking security since it becomes more difficult for a pirate to notice the existence of a watermark or to localize the embedded bits.

In order to devise an effective robust and blind mesh watermarking algorithm in the spectral domain, we have to solve the following two problems:

1. how to achieve the robustness against the connectivity attacks in a blind way;
2. how to obtain the applicability on "big" meshes having more than 10000 vertices.

It seems that for the algorithms that are based on the combinatorial Laplacian mesh spectral analysis [KG00], the most effective solution to the first problem is to use the distribution of a group of spectral coefficients as the watermarking primitive [LB08, LWBL09]. Normally, we cannot embed watermark bits in the individual spectral coefficients obtained after a combinatorial frequency analysis, because they are not individually robust under connectivity changes of the mesh. Instead, the statistical feature (e.g. the distribution) of a large number of spectral coefficients is more or less preserved even after conducting connectivity attacks. Another solution to the first problem is to use a mesh frequency decomposition tool that is capable of producing robust spectral coefficients even under connectivity attacks. It seems that the mesh watermarking research community has been looking for such an analysis tool for a long time, and recently we have found that the manifold harmonics transform proposed by Vallet and Lévy [VL07, VL08] provides a very good robustness of the obtained spectral coefficients. Based on this transform, Liu et al. [LPG08] devised a robust and blind watermarking scheme with a capacity of 5 bits.

Concerning the second problem, i.e. the applicability on "big" meshes, there exist two possible solutions. The first is to segment the "big" mesh into several patches, on which the spectral analysis becomes much less expensive. Note that the segmentation algorithm should be robust against various attacks on the watermarked mesh and

meanwhile does not need the knowledge of the original cover mesh. Rondao-Alface et al. [RAM05, RAMC07] and Luo et al. [LWBL09] have made some efforts in devising such a robust and "blind" mesh segmentation algorithm, which is deemed to be a very difficult problem. The second solution is to use only a very small number of spectral coefficients (often of low frequency) to embed the watermark [CRAS*03, LPG08]. In this way, we only need to compute very few eigenvectors (i.e. spectral decomposition bases) of the established eigen-problem related to the spectral analysis (c.f. Section 3.2.3.2), thus the computational cost is significantly reduced. Note that these two solutions are not mutually exclusive and in fact they can be jointly used so as to further decrease the computational complexity of the watermarking algorithm.

Our objective in this chapter is to use the quantization-based data hiding technique to embed a blind and robust watermark in the manifold harmonics spectral domain of a 3-D mesh. We would like to improve the achievable watermarking capacity in this promising domain (currently 5 bits as in the scheme of Liu et al. [LPG08]), while preserving as well as possible the other performance metrics (i.e. robustness, imperceptibility, security and computational efficiency). In the proposed method, a blind 16-bit watermark is embedded through an iterative SCS quantization of the low frequency coefficients obtained by means of the manifold harmonics analysis.

The remainder of this chapter is organized as follows: the manifold harmonics transform is introduced in Section 7.2; the proposed watermark embedding and extraction algorithms are detailed in Section 7.3; in Section 7.4, some experimental results are presented, with comparisons with the methods of Cho et al. [CPJ07] and Liu et al. [LPG08]; finally, we draw the conclusion in Section 7.5.

7.2 Manifold Harmonics Transform

7.2.1 Formulation

In this subsection, we briefly introduce the manifold harmonics transform and the corresponding inverse transform. Readers could refer to [VL07] and [VL08] for more details about these transforms, especially their derivations.

The objective of the authors of the manifold harmonics transform was to generalize the conventional Fourier analysis to the functions defined on arbitrary 2-manifold surfaces. We know that the Fourier basis functions are the eigenfunctions of the Laplace operator in the Euclidean space. The counterpart of the conventional Laplace operator in the case of 2-manifold space is the Laplace-Beltrami operator, denoted here by Δ_{LB}.

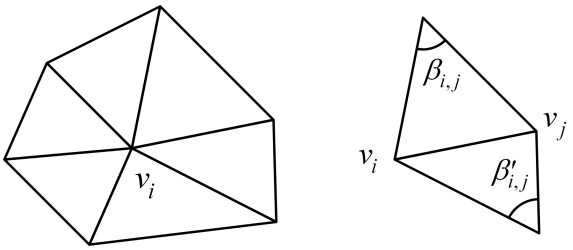


Figure 7.1: This figure illustrates the quantities involved in the determination of the elements of the lumped mass matrix D and the the stiffness matrix Q: (a) $\mathcal{N}_t(i)$ as the set of triangles incident to vertex v_i, (b) $\beta_{i,j}$ and $\beta'_{i,j}$ as the angles opposite to the edge that connects v_i and v_j.

Similar to the Laplace operator in the Euclidean space, Δ_{LB} is defined as the divergence of the gradient for the functions defined over a 2-manifold surface $\mathscr{S}$:

$$\Delta_{LB} = \text{div}\,\text{grad} = \sum_{i,j} \frac{1}{\sqrt{|g|}} \frac{\partial}{\partial \xi_i} \left(\sqrt{|g|} g^{i,j} \frac{\partial}{\partial \xi_j} \right), \tag{7.1}$$

where g is the metric tensor of the surface, $|g|$ denotes the determinant of g, and $g^{i,j}$ are the components of the inverse of g. The eigenfunction and eigenvalue pairs $\left(H^k, \lambda_k\right)$ of the operator Δ_{LB} on $\mathscr{S}$ satisfy the following equation:

$$-\Delta_{LB} H^k = \lambda_k H^k. \tag{7.2}$$

The authors then consider the case where $\mathscr{S}$ is actually a 2-manifold triangular mesh. In this case, the above eigen-problem can be discretized and simplified within the finite element modeling framework on the n vertices $v_i, i \in \{1, 2, ..., n\}$ of the mesh surface $\mathscr{S}$. This yields the following matrix formulation:

$$-Q\mathbf{h}^k = \lambda_k D\mathbf{h}^k, \tag{7.3}$$

where $\mathbf{h}^k = \left[H_1^k, H_2^k, ..., H_n^k\right]^T$, the $n \times n$ matrix D is diagonal and called the lumped mass matrix with $D_{i,i} = \left(\sum_{t \in \mathcal{N}_t(i)} |t|\right)/3$, and Q is also of size $n \times n$ and called the stiffness matrix with

$$\begin{cases} Q_{i,j} = \left(\cot(\beta_{i,j}) + \cot(\beta'_{i,j})\right)/2, \\ Q_{i,i} = -\sum_j Q_{i,j}. \end{cases} \tag{7.4}$$

In the above expressions, $\mathcal{N}_t(i)$ denotes the set of triangles incident to vertex v_i (c.f. Figure 7.1.(a)), $|.|$ gives the area of a triangle, and $\beta_{i,j}, \beta'_{i,j}$ are the two angles opposite to the edge that connects v_i and v_j (c.f. Figure 7.1.(b)).

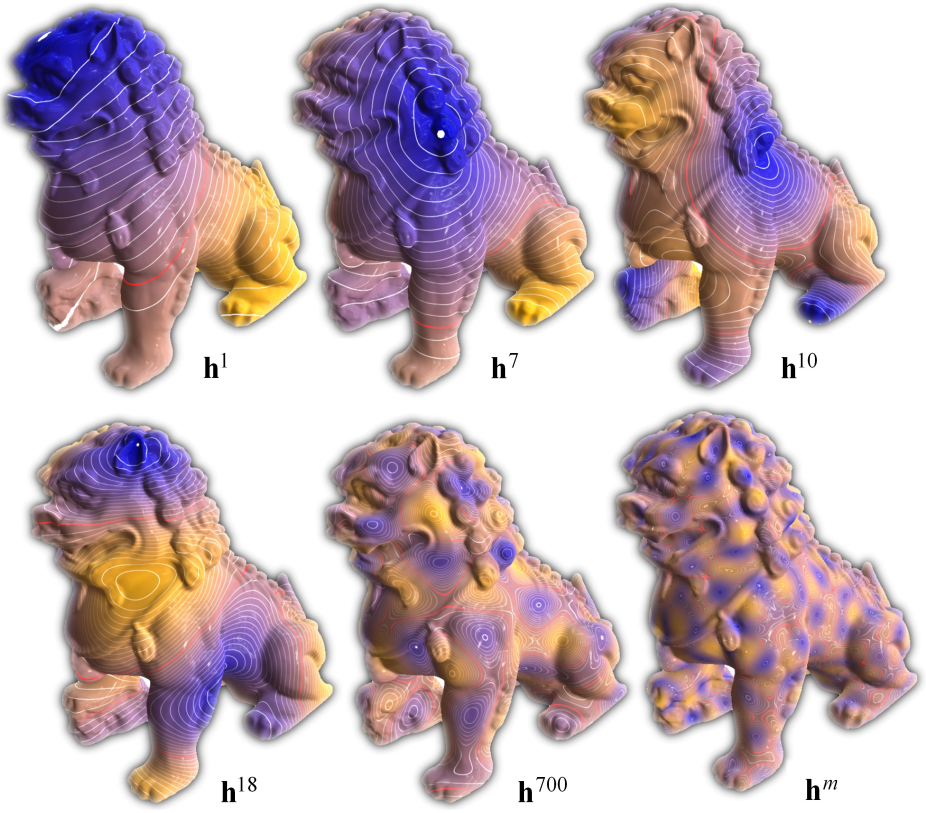


Figure 7.2: Some manifold harmonics bases on the Lion mesh (extracted from [VL07]).

The eigenvectors solutions of Equation (7.3) are the manifold harmonics bases, while the eigenvalues represent the squares of their associated frequencies. The bases are orthogonal with regard to the functional inner product, as described by the following equation:

$$\left\langle \mathbf{h}^k, \mathbf{h}^l \right\rangle_D = \sum_{i=1}^{n} H_i^k D_{i,i} H_i^l = 0, \text{ for } k \neq l. \tag{7.5}$$

We can see that the diagonal elements of the lumped mass matrix D are involved in this functional inner product calculation. We then sort the bases according to the ascending order of their associated frequencies and also scale them so that they all have unit norms under the functional inner product, i.e. we have $\left\|\mathbf{h}^k\right\|_D = \sqrt{\sum_{i=1}^{n} H_i^k D_{i,i} H_i^k} = 1, k \in \{1, 2, ..., n\}$. Figure 7.2, which is extracted from [VL07], illustrates some of the manifold harmonics bases of the Lion mesh on its surface (with iso-value contours). In general, the intrinsic mode of these bases $\mathbf{h}^k, k \in \{1, 2, ..., n\}$ varies from of low frequency to of high frequency as the index k increases.

The spectral coefficients under the manifold harmonics analysis are calculated as the functional inner product between the mesh geometry $\mathbf{x}$ (resp. $\mathbf{y}, \mathbf{z}$) and the sorted and orthonormal bases:

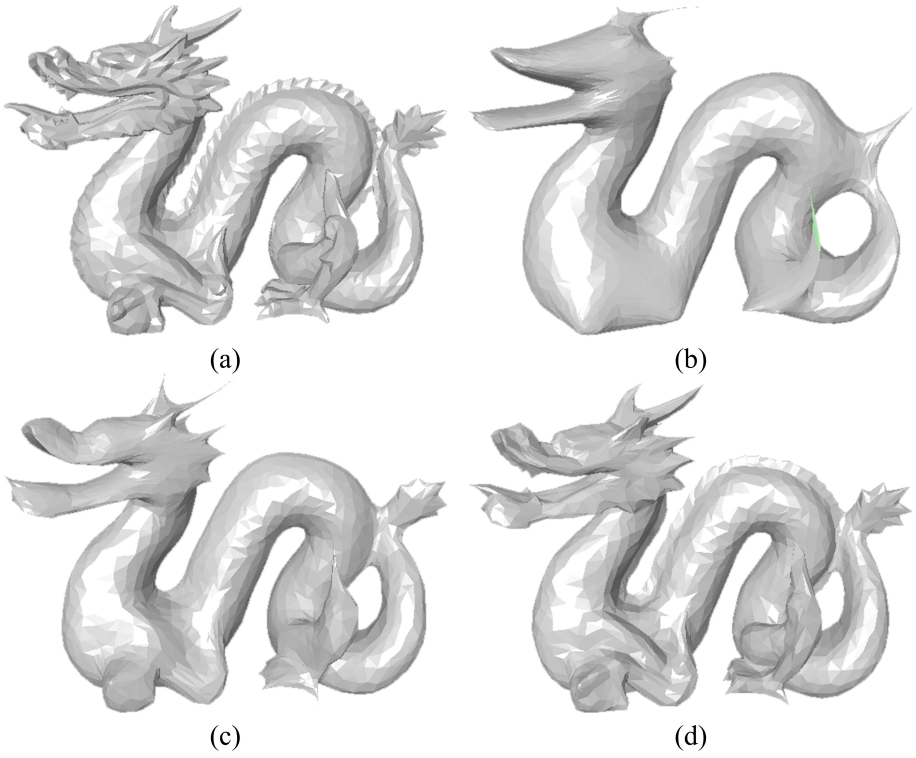


Figure 7.3: Mesh reconstruction with different numbers of manifold harmonics bases and coefficients: (a) the original Dragon having 5205 vertices; (b) the reconstructed Dragon with the first 57 bases; (c) the reconstructed Dragon with the first 274 bases; (d) the reconstructed Dragon with the first 1337 bases.

$$\tilde{x}_k = \left\langle \mathbf{x}, \mathbf{h}^k \right\rangle_D = \sum_{i=1}^{n} x_i D_{i,i} H_i^k. \tag{7.6}$$

Normally, the k-th spectral coefficient amplitude is defined as:

$$c_k = \sqrt{(\tilde{x}_k)^2 + (\tilde{y}_k)^2 + (\tilde{z}_k)^2}. \tag{7.7}$$

The object can be exactly reconstructed by using the inverse manifold harmonics transform. For the geometry $\mathbf{x}$ (resp. $\mathbf{y}, \mathbf{z}$), we have

$$x_i = \sum_{k=1}^{n} \tilde{x}_k H_i^k. \tag{7.8}$$

We can use only the first $m < n$ coefficients and bases to reconstruct an approximated version of the original mesh. Figure 7.3 illustrates the original Dragon mesh and several reconstructed versions with different numbers of involved manifold harmonics bases. It can be seen that by using the first few bases, we can obtain a very smooth mesh possessing the basic shape of the original model. Then, the local geometric details of the mesh are gradually recovered as the number of involved bases increases.

The first few low-frequency manifold harmonics bases (and thus the corresponding

spectral coefficients) can be efficiently calculated by using the band-by-band algorithm [VL07, VL08] combined with an efficient eigen-solver library such as the TAUCS [TCR03] or the SuperLU [DGL09]. For instance, the first 100 coefficients of the Rabbit mesh having about 33.5K vertices can be obtained in less than 40 seconds on an ordinary PC equipped with a 2GHz processor and 2GB memory.

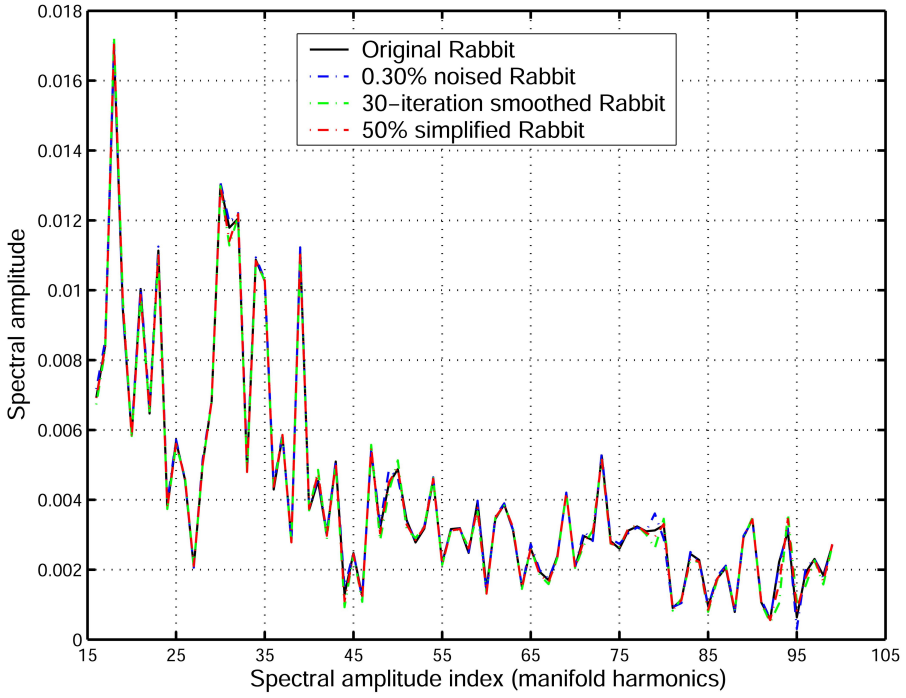
7.2.2 Robustness of the manifold harmonics spectral amplitudes

The robustness of the low-frequency manifold harmonics spectral amplitudes $c_k, k \in \{1, 2, ..., m\}$ is crucial to the performance of our blind watermarking scheme, since we will use these amplitudes as the watermarking primitives. We have experimentally studied and verified this robustness on several mesh models. Figure 7.4.(a) illustrates the curves of the first 100 spectral amplitudes of the original Rabbit mesh and several attacked versions. The conducted attacks are considered to have moderate strengths in the context of mesh watermarking application. We can see that the obtained curves almost coincide, which demonstrates the very strong stability of these low-frequency spectral amplitudes under various geometry and connectivity attacks. On the contrary, as illustrated by Figure 7.4.(b), although the spectral amplitudes in the combinatorial Laplacian analysis are quite robust against geometry attacks, they are vulnerable under connectivity changes.

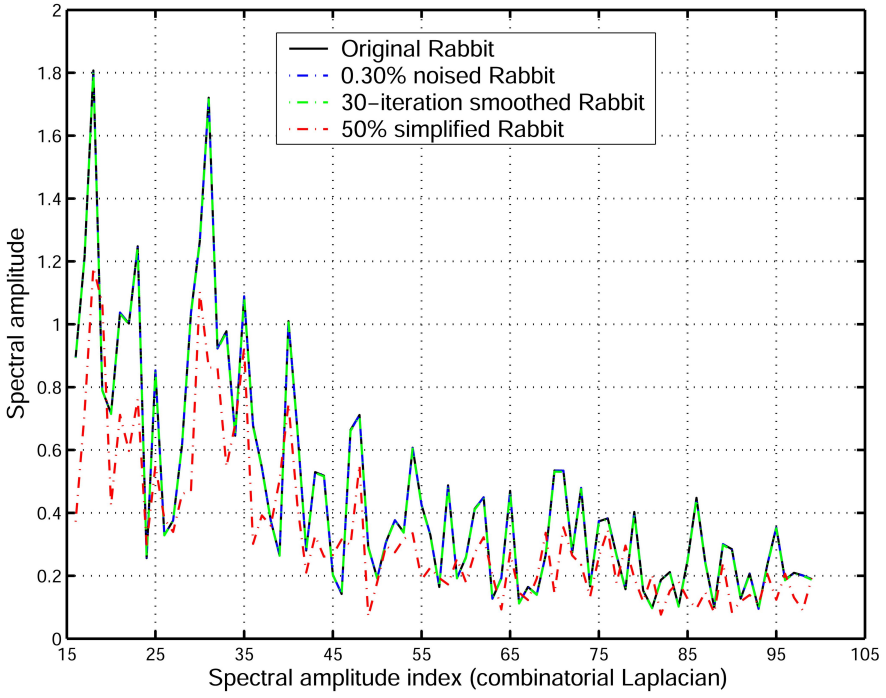
The stability of the first 100 spectral amplitudes has also been quantitatively measured by the following error metric:

$$Err = \frac{\sum_{k=1}^{100} |\hat{c}_k - c_k|}{\sum_{k=1}^{100} |c_k|}, \tag{7.9}$$

where c_k and $\hat{c}_k$ represent respectively the k-th spectral amplitude of the original and attacked models. Table 7.1 provides a quantitative comparison between the stability of the spectral amplitudes in the combinatorial Laplacian analysis and those in the manifold harmonics analysis, in terms of the above relative error metric Err. It is normal that the combinatorial Laplacian has a much better stability under geometry attacks since its spectral analysis bases depend entirely on the mesh connectivity and thus never change under geometry attacks. However, the obvious vulnerability under connectivity attacks makes it nearly impossible to use this analysis for blind and robust mesh watermarking. The manifold harmonics analysis possesses a satisfactory and well balanced overall stability of its spectral amplitudes. Hence, it is well suited for being used in quantization-based blind spectral mesh watermarking.



(a)



(b)

Figure 7.4: The stability of the spectral amplitudes in the manifold harmonics analysis (a) and in the combinatorial Laplacian analysis (b) under different geometry and connectivity attacks. In order to obtain a good visualization of the amplitudes with small values, the first 15 amplitudes, which have very large values, are not plotted.

Table 7.1: Stability comparison of the first 100 spectral amplitudes (on the Rabbit model) in different mesh frequency analyses, in terms of the relative error metric *Err* defined in Equation (7.9). The deformation factor λ is equal to 0.03 for the smoothing attacks.

Attack	Combinatorial Laplacian	Manifold harmonics
0.30% noise	2.63×10^{-4}	1.18×10^{-2}
0.50% noise	4.01×10^{-4}	3.39×10^{-2}
30-iteration smoothing	9.70×10^{-4}	9.38×10^{-3}
50-iteration smoothing	1.62×10^{-3}	1.38×10^{-2}
30% simplification	0.195	2.56×10^{-3}
50% simplification	0.317	5.17×10^{-3}
70% simplification	0.469	1.07×10^{-2}

7.3 Watermark Embedding and Extraction

Our watermark embedding procedure is composed of three main steps:

1. Compute a number of low-frequency manifold harmonics bases of the input mesh and carry out the mesh spectrum decomposition by using Equation (7.6);
2. Quantize some low-frequency spectral amplitudes to hide a 16-bit watermark $w_j, j \in \{1, 2, ..., 16\}$, by using the 2-symbol scalar Costa scheme [EBTG03] (c.f. Chapter 4);
3. Reconstruct the watermarked mesh with the modified coefficients using the inverse manifold harmonics transform as described in Equation (7.8).

In step 2, for each spectral amplitude c_k that is to be watermarked, a structured codebook is given as:

$$\mathcal{U}_{c_k,t_{c_k}} = \bigcup_{l=0}^{1} \left\{ u = zS_{mh} + l\frac{S_{mh}}{2} + t_{c_k}S_{mh}, u \geq 0 \right\}, \tag{7.10}$$

where z is an integer, S_{mh} is the quantization step, $l \in \{0, 1\}$ is the watermark bit from the codeword u, and t_{c_k} is a pseudo-random sequence generated by using a secret key K_{mh}. In order to insert a watermark bit w^{c_k} in c_k, we first find the nearest codeword u_{c_k} to c_k in the codebook that correctly represents w^{c_k}. Then, the quantized spectral amplitude value $c_k^{'}$ is calculated as:

$$c_k^{'} = c_k + \alpha_{mh}\left(u_{c_k} - c_k\right), \tag{7.11}$$

where α_{mh} represents the distortion compensation factor. Finally, the modified spectral coefficients $\tilde{x}_k^{'}$ (resp. $\tilde{y}_k^{'}, \tilde{z}_k^{'}$) can be obtained as:

$$\tilde{x}_k^{'} = \frac{c_k^{'}}{c_k}\tilde{x}_k. \tag{7.12}$$

The quantization mechanism of the spectral amplitudes is quite simple, but there still exist some important details about the whole watermark embedding algorithm. Actually, two issues have to be carefully taken into account during its design: the causality problem and the invariance to similarity transformation.

It can be noticed that once we modify the spectral coefficients of a model from $\tilde{x}_k, \tilde{y}_k, \tilde{z}_k$ to $\tilde{x}_k^{'}, \tilde{y}_k^{'}, \tilde{z}_k^{'}$ and reconstruct a deformed object, the manifold harmonics bases of the reconstructed object are different from those of the original model. It means that after the re-decomposition of the deformed object, the obtained coefficients, denoted by $\tilde{x}_k^{''}, \tilde{y}_k^{''}, \tilde{z}_k^{''}$, are different from the desired values $\tilde{x}_k^{'}, \tilde{y}_k^{'}, \tilde{z}_k^{'}$. One straightforward solution to this causality problem is to repetitively perform the quantization procedure, in a similar way to the strategy adopted by Liu et al. [LPG08]. The previously watermarked mesh is taken as the input to the watermark embedding system in which its spectral amplitudes are re-quantized. This process is carried out for several iterations, until all the bits are correctly embedded. In order to reduce the number of iterations (i.e. the processing time), we take the following measures. First, we do not quantize the first 20 spectral amplitudes (and thus the corresponding spectral coefficients). Experimentally, modifying them will also cause noticeable alterations of the subsequent coefficients and thus increase the iteration number of the watermark embedding procedure. Second, starting from c_{21}, we quantize every 3 out of 4 amplitudes. This creates some "buffering space" between watermarking primitives, which can effectively alleviate the causality problem. Even after taking the above two measures, sometimes we still need a large number of iterations until all the 16 primitive amplitudes are correctly quantized. Therefore, we introduce the third measure: trying to repetitively embed the 16 bits for 3 times. This repetition is capable of reducing the processing time even though we now have more amplitudes ($16 \times 3 = 48$) to quantize. Actually, the 16 bits are considered successfully embedded as long as the majority voting results from the corresponding repetitively embedded bits are correct. The iterative procedure can then be terminated earlier, even when there still exist embedding errors on certain "difficult" amplitudes.

In this paragraph, we present our solution to achieving the invariance to similarity transformations which include translation, rotation and uniform scaling. It can be deduced that under translation, only the first spectral amplitude c_1 is altered. Since c_1 is not involved in the watermark embedding, our method is immune to translation. Meanwhile, it can be easily proven that the manifold harmonics bases are kept unchanged under isometric transformations; therefore, a rotation in the spatial domain x, y, z yields the same rotation in the spectral domain $\tilde{x}_k, \tilde{y}_k, \tilde{z}_k$, without any influence on the coefficient amplitudes c_k. It can also be demonstrated that under a uniform scaling with

a factor s, all the spectral coefficients will be scaled by s^2. In order to be immune to scaling, we determine the quantization step of c_k as $S_{mh} = \beta c_2$, with β a constant. In this way, the codewords in the component-wise SCS codebook of c_k change proportionally with c_k under uniform scaling, so the invariance is ensured. Experimentally, β can be fixed as 0.0015 for all the objects.

Algorithm 7.1 summarizes the whole watermark embedding algorithm. The first 15 steps actually constitute the watermark extraction algorithm. If the embedding algorithm is terminated within few iterations (say less than 6), the procedure can be further continued while neglecting the stop criterion at step 16, in order to get another one or two stego models, which possibly have a smaller err_2 value (c.f. step 15 for its calculation) but a higher induced distortion. Then, we can select one from the obtained stego models as the final watermarked mesh according to the required trade-off between the induced distortion and the robustness. Normally, a model with a smaller err_2 value possesses a stronger robustness.

7.4 Experimental Results and Comparisons

The proposed method has been tested on several meshes such as: Rabbit (33520 vertices), Horse (36043 vertices) and Venus (67173 vertices). Table 7.2 details some statistics about the watermark embedding and extraction on these models. The values in Table 7.2 represent the averages of 5 trials with 5 different random watermark codes. We will first compare our method with the Algorithm I of Cho et al. [CPJ07] (with the same capacity 16 bits and the strength factor α equal to 0.025). The corresponding results of this comparison algorithm are presented in parentheses in Table 7.2. All the tests were carried out on a Pentium IV 2.0GHz processor with 2GB memory. It can be seen that our approach can be successfully applied on large datasets with acceptable embedding and extraction times. The objective and perceptual distortions between the watermarked and original meshes are respectively measured by MRMS and MSDM (c.f. Sections 5.5.1 and 6.6.1). One advantage of our method is that it can introduce relatively high-amplitude deformation while keeping it imperceptible. This point is also confirmed in Figure 7.5, where the original and watermarked meshes, along with the corresponding geometric distance maps, are illustrated. We can hardly observe any visual difference between the cover and stego models.

The robustness of our method has been tested under noise addition, smoothing and simplification, with a comparison with Cho's method. The robustness is evaluated in terms of watermark bit detection ratio (BDR), defined as the ratio of the correctly ex-

Calculate the first 84 spectral coefficients $\tilde{x}_k, \tilde{y}_k, \tilde{z}_k$ and their amplitudes c_k of the input mesh by using the manifold harmonics analysis;
Record the difference part of the geometry $\bar{x}_i = x_i - \sum_{k=1}^{84} \tilde{x}_k H_i^k$ (resp. $\bar{y}_i, \bar{z}_i$);
Initialization: $k = 21, j = 1$;
while $k \leq 84$ **do**
  if $k\%4 \neq 0$ (not being "buffering space") **then**
    Construct the 2-symbol SCS codebook $\mathcal{U}_{c_k,t_{c_k}}$ for c_k with the quantization step $S_{mh} = \beta c_2$;
    Find the nearest codeword u_{c_k} in $\mathcal{U}_{c_k,t_{c_k}}$ to c_k, and record the represented bit of u_{c_k} as $\tilde{w}_j$;
    $j \leftarrow j + 1$;
  end if
  $k \leftarrow k + 1$;
end while
for $j = 1$ to 16 **do**
  Deduce the extracted bit $\hat{w}_j$ through majority voting between $\tilde{w}_j, \tilde{w}_{j+16}, \tilde{w}_{j+32}$;
end for
Compare with $w_j, j \in \{1, 2, ..., 16\}$ and its periodic extension $w_j, j \in \{1, 2, ..., 48\}$ (for $j > 16$ we have $w_j = w_{(j\%16)}$), count the bit error number of $\hat{w}_j, j \in \{1, 2, ..., 16\}$ as err_1 and the bit error number of $\tilde{w}_j, j \in \{1, 2, ..., 48\}$ as err_2;
if $(err_1 == 0)$ AND $(err_2 < 6)$ **then**
  Stop the iteration, take the current mesh as the watermarked model;
else
  Initialization: $k = 21, j = 1$;
  while $k \leq 84$ **do**
    if $k\%4 \neq 0$ **then**
      Deduce the to-be-embedded bit of c_k as $w_{(j\%16)}$;
      Calculate the quantized value $c_k^{'}$ by using the 2-symbol scalar Costa scheme with quantization step $S_{mh} = \beta c_2$;
      Obtain the modified spectral coefficients $\tilde{x}_k^{'}, \tilde{y}_k^{'}, \tilde{z}_k^{'}$ by using Equation (7.12);
      $j \leftarrow j + 1$;
    end if
    $k \leftarrow k + 1$;
  end while
end if
Reconstruct the deformed geometry $\mathbf{x}^{'}$ with $x_i^{'} = \bar{x}_i + \sum_{k=1}^{84} \tilde{x}_k^{'} H_i^k$ (resp. $\mathbf{y}^{'}, \mathbf{z}^{'}$);
Return to step 1 with the reconstructed model as input.

Algorithm 7.1: Robust and blind spectral mesh watermark embedding procedure.

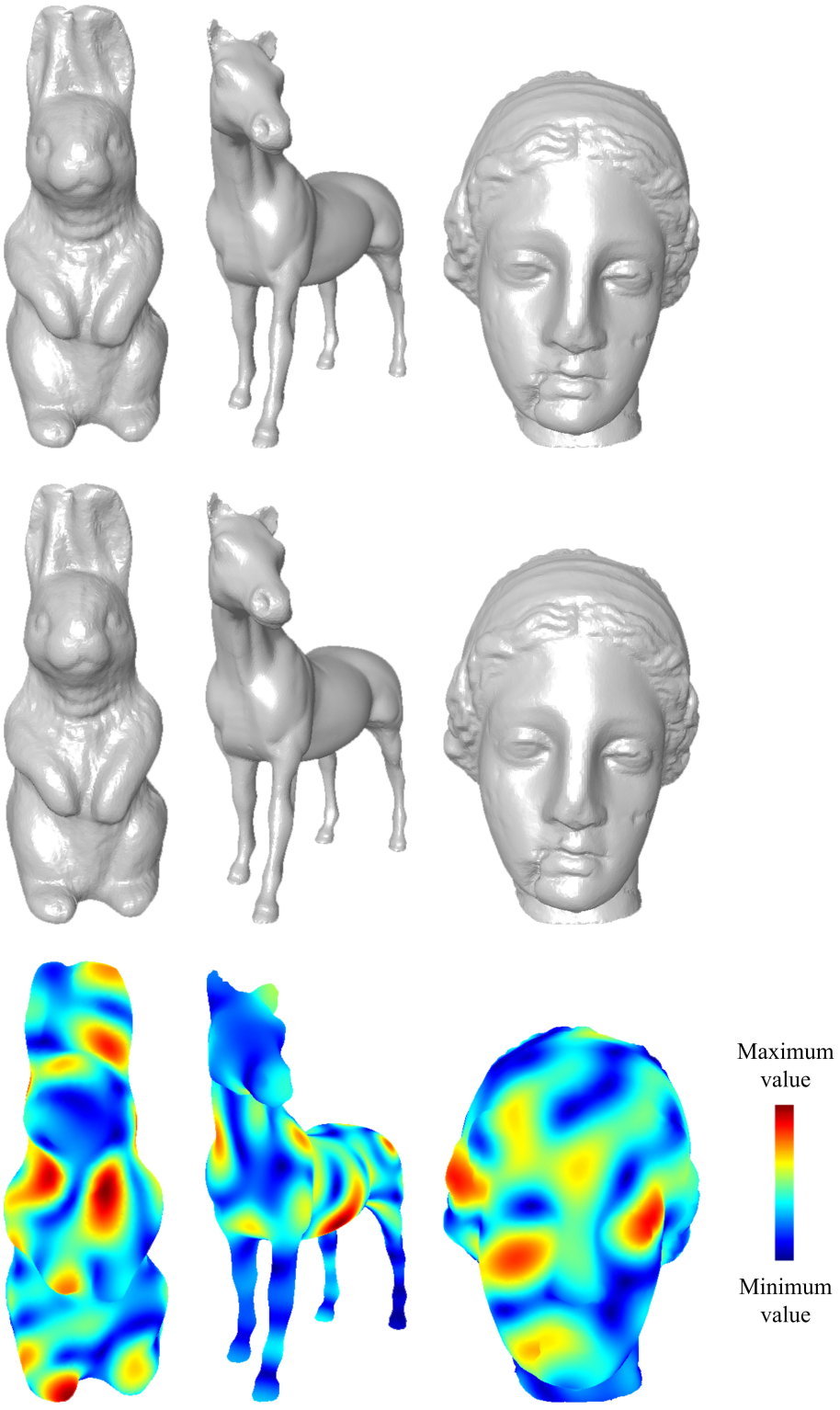


Figure 7.5: In the first row are the original cover meshes; in the second row are illustrated the watermarked models; the corresponding geometric distance maps are shown in the last row.

Table 7.2: Baseline evaluations of the proposed method (16 bits capacity, in the parentheses are the results of Cho's method).

Model	Rabbit	Horse	Venus
Iteration number	12.2	13.4	8.6
Embedding time (s)	316.6 (1.5)	278.0 (1.6)	521.0 (2.4)
Extraction time (s)	23.5 (< 1.0)	20.1 (< 1.0)	52.9 (< 1.0)
MRMS by WM (10^{-3})	2.37 (2.14)	2.37 (1.57)	1.95 (1.19)
MSDM by WM	0.17 (0.20)	0.13 (0.21)	0.13 (0.21)

Table 7.3: Robustness evaluation results in terms of BDR (16 bits capacity, in the parentheses are the results of Cho's method).

Noise	0.10%	0.20%	0.30%	0.40%	0.50%
Rabbit	1 (1)	0.99 (1)	0.99 (0.96)	0.93 (0.94)	0.81 (0.93)
Horse	0.98 (1)	0.96 (0.98)	0.89 (1)	0.80 (0.99)	0.74 (0.96)
Venus	1 (1)	0.99 (0.93)	0.96 (0.91)	0.79 (0.88)	0.79 (0.84)
Smoothing	10-iteration	20-iteration	30-iteration	40-iteration	50-iteration
Rabbit	0.98 (0.98)	0.94 (0.96)	0.91 (0.93)	0.93 (0.90)	0.88 (0.90)
Horse	0.90 (1)	0.81 (0.99)	0.70 (0.99)	0.63 (0.98)	0.59 (0.94)
Venus	0.99 (0.96)	0.95 (0.89)	0.89 (0.83)	0.83 (0.79)	0.79 (0.71)
Simplification	10%	30%	50%	70%	90%
Rabbit	1 (0.98)	0.99 (0.90)	0.94 (0.76)	0.90 (0.71)	0.68 (0.64)
Horse	0.99 (0.91)	0.93 (0.48)	0.84 (0.81)	0.58 (0.63)	0.53 (0.71)
Venus	1 (0.91)	0.98 (0.90)	0.99 (0.88)	0.81 (0.79)	0.51 (0.64)

tracted bits. Table 7.3 presents the robustness evaluation results of both methods (those of Cho's method are in parentheses), which are also the averages of 5 trials. Our method is quite robust against various attacks, as long as they do not significantly modify the shape of the watermarked mesh. However, the results are not that good on Horse. For this object we guess that the manifold harmonics bases may be sensitive to the deformation of the obtrusive parts of this model (e.g. the ears and the feet) under attacks.

Under the current parameter setting, the two methods show comparable overall robustness against various attacks, except for extremely strong ones, against which Cho's method is more resistant. More precisely, it seems that our method works better for mesh simplification while Cho's method performs better for noise addition and smoothing. Under this comparable robustness premise, the watermarked models of Cho's method have lower objective geometric distortion, while ours have a higher visual quality (c.f. the MRMS and MSDM values presented in Table 7.2). Actually, Cho's method is prone to introduce some ring-like high-frequency distortions to the watermarked meshes, especially on smooth regions like the cheek of Venus (c.f. Figure 7.6.(a)). Contrarily, the distortion induced by our method is of low frequency (c.f. the geometric distance maps illustrated in the last row of Figure 7.5) and it is difficult for the human visual system to perceive it. Our method is also more secure, both in the general sense

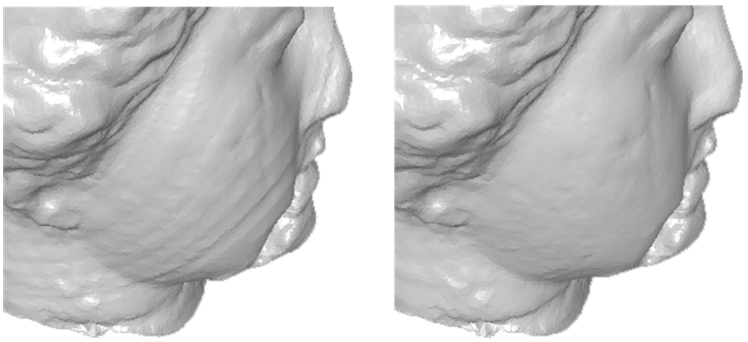


Figure 7.6: Watermark imperceptibility comparison: (a) Venus watermarked by Cho's method; (b) Venus watermarked by our method.

(e.g. to prevent a pirate from suspecting that the current mesh is watermarked) and in the information-theoretic sense [PFCPG05]. Compared to Cho's method, the main drawback of our scheme is its low achievable capacity. The improvement on this point constitutes one part of our future work.

In the following, we compare the proposed method with the method of Liu et al. [LPG08] that is also based on the manifold harmonics transform (with a capacity of 5 bits). In order to perform a fair comparison between the two schemes, we have slightly modified our method so as to ensure the premise of a same watermarking capacity. More precisely, 5 bits are embedded repetitively for 5 times from the 21st to the 70th spectral amplitudes (therefore we quantize every one out of two amplitudes). Meanwhile, in Liu's algorithm [LPG08], the 21st to the 70th amplitudes are divided into 5 slots and one bit is embedded in each slot by modifying the relative relationship between one selected amplitude and the average of the other amplitudes in the slot. The parameters of these two algorithms are chosen so that they have roughly a comparable overall robustness performance. For Liu's method, we use the option of normal slot combined with non-aggressive embedding so as to introduce less distortion (the parameter s in their algorithm is fixed as 0.20).

Table 7.4 presents the comparison results concerning the computational complexity and the induced distortion, and Table 7.5 presents the robustness evaluation results of the two algorithms, also in terms of BDR. In both tables, the results of Liu's method are presented in parentheses and all the values are the averages of 3 trials. It can be seen that with a roughly comparable robustness level, Liu's method introduces higher geometric and perceptual distortions than our method. This implies that our watermarking scheme has a better trade-off between the robustness and the induced distortion.

We have found that the induced distortion in Liu's method depends heavily on the

Table 7.4: Baseline evaluations of the proposed method (5 bits capacity, in the parentheses are the results of Liu's method).

Model	Rabbit	Horse	Venus
Iteration number	7.3 (6.0)	6.7 (8.3)	4.7 (5.7)
Embedding time (s)	139.6 (116.6)	107.6 (140.0)	208.7 (250.6)
Extraction time (s)	18.2 (18.9)	15.4 (16.1)	43.2 (43.4)
MRMS by WM (10^{-3})	1.50 (3.32)	1.43 (4.42)	1.14 (2.72)
MSDM by WM	0.11 (0.20)	0.09 (0.20)	0.09 (0.12)

Table 7.5: Robustness evaluation results in terms of BDR (5 bits capacity, in the parentheses are the results of Liu's method).

Noise	0.10%	0.20%	0.30%	0.40%	0.50%
Rabbit	1 (1)	1 (1)	0.93 (0.93)	0.87 (0.87)	0.93 (0.93)
Horse	1 (1)	0.93 (0.87)	0.93 (0.93)	0.93 (0.93)	0.73 (0.73)
Venus	1 (1)	0.93 (0.93)	0.87 (0.87)	0.73 (0.80)	0.80 (0.80)
Smoothing	10-iteration	20-iteration	30-iteration	40-iteration	50-iteration
Rabbit	1 (0.80)	1 (0.73)	1 (0.67)	1 (0.60)	0.93 (0.60)
Horse	1 (0.93)	0.93 (0.87)	0.73 (0.80)	0.67 (0.80)	0.63 (0.80)
Venus	1 (0.80)	1 (0.80)	1 (0.80)	0.87 (0.80)	0.73 (0.73)
Simplification	10%	30%	50%	70%	90%
Rabbit	1 (1)	1 (1)	1 (0.93)	1 (0.87)	0.73 (0.67)
Horse	1 (1)	1 (0.80)	0.93 (0.73)	0.93 (0.73)	0.40 (0.73)
Venus	1 (1)	1 (1)	1 (1)	1 (0.93)	0.93 (0.73)

relationship between the initially represented bits of the slots and the embedded watermark code. The distortion becomes very big when there exist many slots in the mesh (say greater than or equal to 3) whose initially represented bits are different from the watermark bits to be embedded. For instance, on the Venus model, if there exist 4 slots whose represented bits are incorrect with regard to the watermark bits, then the MRMS distortion attains 5.72×10^{-3}. Mainly due to this fact, it seems very difficult to increase the capacity of their method while keeping an acceptable amount of distortion. Considering this distortion issue, for the experiments of Liu's method, we have explicitly designated the watermark bits to be embedded in the 3 trials. The 5-bit watermarks have respectively 1, 3 and 4 bits different from the initially represented bits of the slots. On the contrary, our method can achieve a higher capacity (currently until 16 bits) while not seriously increasing the distortion, and meanwhile the induced distortion of our scheme is almost independent from the embedded watermark bits.

7.5 Conclusion

A new blind and robust spectral mesh watermarking method has been proposed in this chapter. A 16-bit watermark is embedded in a mesh by iteratively quantizing its

low frequency spectral amplitudes obtained after a manifold harmonics transform. The main features of our method are its high imperceptibility, its good robustness against connectivity attacks, and its applicability to large meshes. However, our method may fail for certain objects: either the model's manifold harmonics spectral coefficients are not that robust, or it is impossible to correctly embed the watermark, even after many iterations. It would be interesting to investigate the reasons for these two phenomena. We are also interested in improving the watermark robustness and capacity, and in devising an efficient and elegant way to solve the causality problem.

Chapter 8

A Benchmark for Robust Mesh Watermarking

Contents

This chapter presents a benchmark for the evaluation of robust mesh watermarking methods. It comprises a data set, a software tool and two evaluation protocols. The data set contains several "standard" mesh models on which we suggest to test the watermarking algorithms. The software tool integrates both objective and perceptual measurements of the distortion induced by watermark embedding, and also the implementation of a variety of attacks on watermarked meshes. Besides, two different application-oriented evaluation protocols are proposed, which define the main steps to follow when conducting the evaluation experiments. The robust and blind algorithms

described in the previous chapters, as well as the method of Cho et al. [CPJ07], are tested and compared by using the proposed benchmarking framework.

A paper which describes this benchmarking work has been prepared and is now to be submitted [WLDB09a].

8.1 Motivation and Contributions

When a new robust mesh watermarking scheme is proposed, we often want to compare it with some existing methods so as to fairly access its strong and weak points. However, at present, it seems difficult and time-consuming to carry out such a comparison, mainly because the authors of different methods often use different mesh models, distortion metrics, attacks and evaluation methodologies when reporting their experimental results. This problem can also be observed in the last three chapters when presenting the experimental results of our robust mesh watermarking algorithms. For instance, the used mesh models are different in the three chapters which describe respectively the wavelet-based, moment-based and spectral-transform-based methods; the Laplacian smoothing attacks in these chapters are with different deformation factor values; the used robustness evaluation metrics (normalized correlation, bit error rate and bit detection ratio) are different for the three methods; and finally there does not exist a clear and consistent strategy when comparing our schemes with the other methods. The main negative consequence of this situation is that we have to re-implement and/or re-test the existing methods when carrying out the experimental comparisons, just as what we have done in the last three chapters. This re-implementation and/or re-testing constitutes an extra (heavy) burden of the mesh watermarking researchers and may also lead to some unreliable comparison results. Hence, our objective in this chapter is to construct a benchmarking software tool for the evaluation of robust mesh watermarking algorithms and also to introduce two application-oriented performance assessment protocols, so as to facilitate the experimental comparisons between different schemes.

Indeed, it is almost impossible to assess the performance of a watermarking algorithm completely through theoretical analysis. Therefore, researchers often have to rely on a benchmarking system combined with a commonly used protocol to conduct an experimental evaluation of their algorithms. Several benchmarking tools and protocols have been proposed for image watermarking evaluation, such as Stirmark [PAK98, Pet00], Checkmark [PVM*01] and Optimark [STN*01]. Contrarily, to the best of our knowledge, the benchmarking of 3-D mesh watermarks was only addressed by Bennour and Dugelay [BD07]. They propose to use some existing software tools to measure the objective distance between cover and stego models, and to perform attacks on watermarked meshes. The authors also propose a formula to calculate a final score as the robustness evaluation result and suggest a four-element structure to report the overall performance of a robust mesh watermarking scheme. Compared to their proposal, our contributions are threefold:

1. We provide a publicly available dataset collection of 3-D mesh models and a software tool for the purpose of mesh watermark evaluation (available at `http://liris.cnrs.fr/meshbenchmark/`).
2. Two protocols are defined for the capacity-distortion-robustness evaluation. In this way, researchers only need to provide some brief tables to report the performance of their watermarking schemes. The comparison then becomes easy and reliable since we all use the same models, distortion metrics and robustness evaluation methodologies.
3. Our wavelet-based (c.f. Chapter 5), moment-based (c.f. Chapter 6) and spectral-domain-based (c.f. Chapter 7) robust mesh watermarking algorithms, as well as the method of Cho et al. [CPJ07], are compared by using the proposed benchmarking software and protocols. The procedure of this comparison demonstrates that our evaluation framework is easy to use and also very effective. It is worthwhile pointing out that the most important difference between the experiments in this chapter and those in the last three chapters is that we now use the same models, attacks, metrics, and most importantly the same evaluation methodologies to compare the different methods within a systematic framework.

The remainder of this chapter is organized as follows: Section 8.2 introduces the evaluation targets of our mesh watermarking benchmark; Sections 8.3 and 8.4 present respectively the distortion metrics and the attacks integrated in our benchmarking software tool; we propose two different application-oriented evaluation protocols in Section 8.5; the evaluation results of some recent algorithms, obtained by using the proposed benchmark, are presented in Section 8.6; finally, we draw the conclusion in Section 8.7.

8.2 Evaluation Targets

As mentioned in Section 2.2.2, a robust watermarking scheme is often evaluated in four different aspects: *capacity*, *distortion*, *robustness* and *security*. The *capacity* is the number of bits of the hidden message conveyed by the watermark. The *distortion* measures the difference between the original cover content and its watermarked version. Note that this induced distortion can be measured either objectively or perceptually. The *robustness* indicates how resistant the watermarking scheme is against various routine operations on the watermarked content. A *secure* watermarking scheme should be able to withstand the malicious attacks that aim to break down the whole watermarking-based copyright protection system through, for instance, secret key disclosure or inversion of the watermark embedding procedure. In the proposed mesh watermarking benchmark, we

only considers the capacity, distortion and robustness evaluations, while discarding the security metric. The main reason is that the research on 3-D mesh watermarking is still in its early stage (c.f. Chapter 3) and until now the community has been interested in achieving robustness against connectivity attacks while paying little attention on the security, a rather high-level requirement. Finally, when reporting the evaluation results, the authors should also indicate whether their scheme is blind, semi-blind or non-blind.

Practically, in order to evaluate a robust mesh watermarking scheme by using the above metrics, we need a well-defined protocol that indicates the steps to follow when conducting the experiments. Before presenting our application-oriented evaluation protocols in Section 8.5, we will first explain how we measure the distortion induced by the mesh watermark embedding procedure and the various attacks against which we would like to test the robustness.

8.3 Distortion Metrics

The watermark embedding process introduces some amount of distortion to the original cover mesh. This distortion can be measured either *objectively* or *perceptually*. For the objective measurement, we propose to use the maximum root mean square error (MRMS). In general, the root mean square error (RMS) from one 3-D surface $\mathcal{S}$ to another 3-D surface $\mathcal{S}^{'}$ is defined as:

$$d_{RMS}(\mathcal{S},\mathcal{S}^{'}) = \sqrt{\frac{1}{|\mathcal{S}|}\int\int_{p\in\mathcal{S}} d(p,\mathcal{S}')^2 d\mathcal{S}}, \tag{8.1}$$

where p is a point on surface $\mathcal{S}$, $|\mathcal{S}|$ is the area of $\mathcal{S}$, and $d(p,\mathcal{S}^{'})$ denotes the point-to-surface distance between p and $\mathcal{S}^{'}$. This RMS distance is not symmetric and generally we have $d_{RMS}(\mathcal{S},\mathcal{S}^{'}) \neq d_{RMS}(\mathcal{S}^{'},\mathcal{S})$. Therefore, we can define the MRMS distance between a cover mesh $\mathcal{M}$ and its watermarked version $\mathcal{M}^{'}$ as:

$$d_{MRMS}(\mathcal{M},\mathcal{M}^{'}) = \max\left(d_{RMS}(\mathcal{M},\mathcal{M}^{'}), d_{RMS}(\mathcal{M}',\mathcal{M})\right). \tag{8.2}$$

Different from the simple vertex-to-vertex distance metrics (e.g. the vertex coordinates PSNR), MRMS measures the surface-to-surface distance between two meshes. The distortion measured by MRMS is more accurate, especially when the two meshes under comparison do not have the same connectivity. Moreover, in this way, we can easily and accurately compare the distortion induced by the watermark embedding and that caused by a connectivity attack, with the usage of a same geometric distance metric. The calculation of MRMS has been implemented in some free software tools such as Metro [CRS98] and MESH [ASCE02]. We included the implementation of Metro in our

benchmarking software (We are very grateful to Dr. P. Cignoni for allowing us to carry out this integration).

However, it is well known that the objective surface-to-surface distances, such as the MRMS, do not correctly reflect the visual difference between two meshes [LGD*06, CGEB07]. Thus, we need a perceptual metric to measure the visual distortion induced by the watermark embedding. For this purpose, we have considered the mesh structural distortion measure (MSDM) proposed by Lavoué et al. [LGD*06], and have integrated it in the benchmarking software. This metric follows the concept of structural similarity recently introduced by Wang et al. [WBSS04] for 2-D image quality assessment, and well reflects the perceptual distance between two 3-D objects. The local MSDM distance between two mesh local windows p and q (respectively in $\mathcal{M}$ and $\mathcal{M}'$) is defined as follows:

$$d_{LMSDM}(p,q) = \left(0.4 \times L(p,q)^3 + 0.4 \times C(p,q)^3 + 0.2 \times S(p,q)^3\right)^{\frac{1}{3}}, \tag{8.3}$$

where L, C and S represent respectively curvature, contrast and structure comparison functions:

$$L(p,q) = \frac{\left\|\mu_p - \mu_q\right\|}{\max(\mu_p, \mu_q)}, \tag{8.4}$$

$$C(p,q) = \frac{\left\|\sigma_p - \sigma_q\right\|}{\max(\sigma_p, \sigma_q)}, \tag{8.5}$$

$$S(p,q) = \frac{\left\|\sigma_p\sigma_q - \sigma_{pq}\right\|}{\sigma_p\sigma_q}, \tag{8.6}$$

with μ_p, σ_p and σ_{pq} respectively the mean, standard deviation and covariance of the curvature over the mesh local windows. The global MSDM measure between two meshes $\mathcal{M}$ and $\mathcal{M}'$, is defined as a Minkowski sum of their n local window distances:

$$d_{MSDM}(\mathcal{M}, \mathcal{M}') = \left(\frac{1}{n}\sum_{j=1}^{n} d_{LMSDM}(p_j, q_j)^3\right)^{\frac{1}{3}} \in [0,1). \tag{8.7}$$

Its value tends toward 1 (theoretical limit) when the measured objects are visually very different and is equal to 0 for identical ones. The main reasons for choosing this perceptual distortion metric are its strong robustness and its high correlation with the subjective evaluation results given by human beings [LGD*06].

8.4 Attacks

In general, there are three kinds of routine attacks on a watermarked mesh (c.f. Section 3.3): *file attack*, *geometry attack* and *connectivity attack*. In the following, we will give

examples for each kind of attacks and present the corresponding implementations in our benchmarking software. In the proposed software, the types and parameters of the attacks performed on a given mesh can be adjusted through a "standardized" configuration file. Moreover, the user can choose whether he/she would like to calculate the objective and/or the perceptual distortions induced by the performed attacks.

8.4.1 File attack

This attack consists in reordering the vertices and/or the facets in the mesh file, and it does not introduce any modification to the mesh shape. A robust mesh watermark should be perfectly invariant to this kind of attack. When carrying out the file attack, the benchmarking software uses a randomly selected key to rearrange the vertex and facet indices in their corresponding lists in the mesh file.

8.4.2 Geometry attack

In a geometry attack, only the vertex coordinates are modified while the mesh connectivity is kept unchanged. Our benchmarking software comprises the following geometry attacks.

Similarity transformation

This operation includes translation, rotation, uniform scaling and their combination. Like the above vertex/facet reordering operation, the similarity transformation always keeps the mesh shape intact. Actually, these two kinds of operations are jointly called content-preserving attacks (c.f. Section 3.2.1), through which a robust watermark, or even a fragile watermark, should be able to survive. In our implementation, in each run of the similarity transformation, the watermarked mesh is successively subject to a random translation, a random rotation and a random uniform scaling. Figure 8.1.(b) illustrates a transformed version of the original Stanford Bunny model that is shown in Figure 8.1.(a).

Noise addition

This attack aims to simulate the artifacts introduced during mesh generation and the errors induced during data transmission. We propose to add pseudo-random noises on vertex coordinates x_i according to the following equation (resp. y_i, z_i):

$$x_i^{'} = x_i + a_i.\bar{d}, \tag{8.8}$$

where $\bar{d}$ denotes the average distance between the mesh vertices and the mesh center,

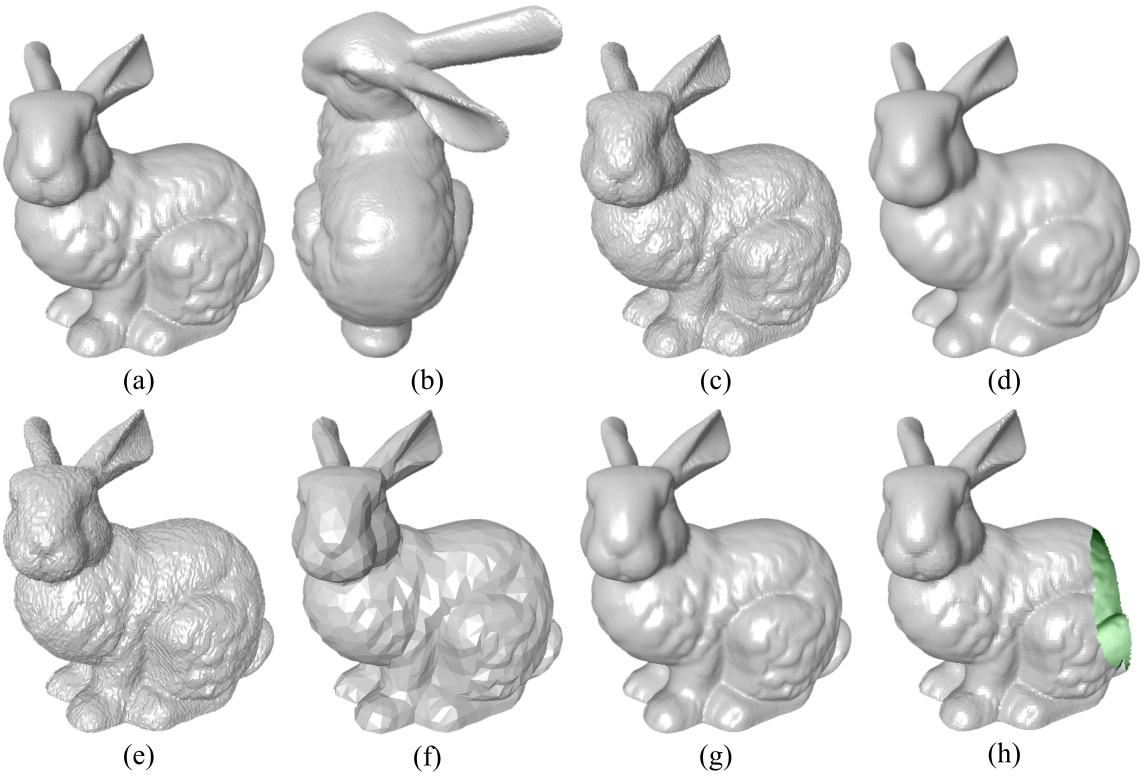


Figure 8.1: The Stanford Bunny model and seven attacked versions: (a) the original mesh having 34835 vertices and 104499 edges; (b) after similarity transformation; (c) after noise addition ($A = 0.30\%$); (d) after Laplacian smoothing ($\lambda = 0.10, N_{itr} = 30$); (e) after vertex coordinates quantization ($R = 8$); (f) after simplification ($E_{sim} = 95\%$); (g) after subdivision (1 iteration, Loop scheme); (h) after cropping ($V_{cr} = 10\%$).

and a_i is the noise strength for x_i. The mesh center is calculated by using the analytic and continuous volume moments of the mesh [ZC01, TSV03], which is much more robust than simply calculating it as the average position of the mesh vertices (c.f. Section 6.4.1). This robust mesh center calculation ensures a same level of induced distortion when the watermark embedding changes the mesh connectivity or when the noise addition is combined with a connectivity modification. a_i is a pseudo-random number uniformly distributed in interval $[-A, A]$, with A the maximum noise strength. Figure 8.1.(c) illustrates a noised version of the Stanford Bunny mesh.

Smoothing

Surface smoothing is a common operation used to remove the noises introduced during the mesh generation process through 3-D scanning. For the purpose of mesh watermark benchmarking, we choose to use a Laplacian smoothing [Tau00] with different iteration numbers N_{itr} while fixing the deformation factor λ as 0.10. Figure 8.1.(d) shows a smoothed Bunny model.

Vertex coordinates quantization

This operation is largely used in lossy mesh compression. Under a R-bit uniform quantization, the x_i (resp. y_i, z_i) coordinate of each vertex is rounded to one of the 2^R eligible quantized levels. Figure 8.1.(e) illustrates a Bunny model whose vertex coordinates are quantized.

8.4.3 Connectivity attack

In a connectivity attack, the mesh connectivity information, i.e. the adjacency relationship between vertices, is changed. Meanwhile, the coordinates of the original vertices may also be modified, such as in some of the surface simplification and subdivision schemes. We have implemented the following connectivity attacks in the software tool.

Simplification

The original version of a mesh model (especially the one obtained by a 3-D scan) often possesses a very high complexity, sometimes with more than 1 million vertices. This high complexity is necessary to ensure a good precision. In practical applications, the watermark is often embedded in the original complex model, and then the model is simplified so as to adapt to the capacity of the available resources. In the benchmarking software we integrated the mesh simplification algorithm of Lindstrom and Turk [LT98], which provides a good trade-off between the precision of the simplified model and the computational efficiency. The user can designate the edge reduction ratios E_{sim} of the simplification operations. Figure 8.1.(f) shows a simplified Bunny model.

Subdivision

In this operation, vertices and edges are added to the original mesh to obtain a modified version that is normally smoother and of a higher visual quality. The watermark robustness is tested against three typical subdivision schemes, always with one iteration step: the simple midpoint scheme, the $\sqrt{3}$ scheme and the Loop scheme [ZS00]. Note that the midpoint scheme adds vertices in the middle of the existing edges, and also edges within the existing facets. This subdivision scheme, which may be performed by a pirate as an attack, does not introduce any distortion to the test model; therefore, ideally a robust mesh watermark should be invariant to it. Figure 8.1.(g) illustrates a subdivided Bunny model.

Cropping

In this attack, one part of the watermarked mesh is cut off and thus lost. This at-

tack could happen when we create a new model by combining parts extracted from several other objects. We propose to conduct the copping attacks with different approximative vertex cropping ratios V_{cr}. In our implementation, for each cropping ratio, 3 attacked models are generated. These models are obtained by cutting the original stego mesh along 3 randomly selected orthogonal axes. Figure 8.1.(h) shows a cropped Bunny model.

Finally, it is important to repeat the attacks with a random nature (i.e. file attack, similarity transformation, noise addition and cropping), for at least 3 times, in order to ensure the reliability of the obtained robustness evaluation results.

8.5 Evaluation Protocols

The objective of a watermark evaluation protocol is to define the main steps to follow when conducting the experimental assessment of a watermarking scheme. In the case of image watermarking, the authors of Stirmark [PAK98] propose to first fix the watermark capacity at about 70 bits and also to limit the induced distortion to be less than 38 dB in terms of PSNR. After that, Stirmark system carries out a series of attacks on the watermarked image. Then, the user tries to extract watermarks from the obtained attacked stego images. Finally, several plots or tables are reported, which basically indicate the robustness metric (e.g. message error rate) versus the amplitudes of the different kinds of attacks.

We define here two similar protocols for the evaluation of robust mesh watermarking schemes. We call the first protocol *perceptual quality oriented* and the second one *geometric quality oriented*. The motivation of establishing two different protocols is that different mesh-based applications have very different restrictions on the objective and perceptual distortions induced by the watermark embedding. For example, for the meshes used in digital entertainment, we should first of all ensure that the induced distortion is not annoying to human eyes (i.e. the watermarked model should have a very high visual quality), while the amount of induced objective distortion is less important. On the contrary, for the meshes used in computer-aided design and medical imaging, it is often required that the objective distortion should be very small, while the visual quality of the watermarked model is relatively less important.

The perceptual quality oriented evaluation protocol consists of the following steps:

1. Embed a watermark W in a test mesh $\mathcal{M}$ by using a secret key K to obtain a watermarked model $\mathcal{M}^{'}$; make sure that the induced perceptual distortion $d_{MSDM} \leq$

Table 8.1: Attacks used in the evaluation protocols.

Attack	Parameter	Parameter values
File attack	times	3
Similarity transformation	times	3
Noise addition*	A	0.05%, 0.10%, 0.30%, 0.50%
Smoothing ($\lambda = 0.10$)	N_{itr}	5, 10, 30, 50
Quantization	R	11, 10, 9, 8, 7
Simplification	E_{sim}	10%, 30%, 50%, 70%, 90%, 95%, 97.5%**
Subdivision (1 iteration)	scheme	midpoint, $\sqrt{3}$, Loop
Cropping	V_{cr}	10%, 30%, 50%

* For each noise amplitude, it is necessary to repeat 3 times.
** The ratio 97.5% is only for large meshes having $\geq$ 100K vertices.

0.20 and the induced objective distortion $d_{MRMS} \leq 0.08\%.l_{bbd}$, where l_{bbd} denotes the diagonal length of the mesh's bounding box.

2. Carry out the suggested attacks listed in Table 8.1 on the stego mesh $\mathcal{M}'$, by using the implemented benchmarking software.
3. Try to extract/detect the embedded watermark W from the obtained attacked stego models and record the extraction/detection robustness evaluation results.
4. Repeat steps 1-3 for 5 times with different randomly selected watermark sequences and secret keys.
5. Repeat steps 1-4 for each test mesh from the standard dataset collection available at the benchmark website.

The two selected distortion thresholds in the perceptual quality oriented protocol (i.e. 0.20 for d_{MSDM} and $0.08\%.l_{bbd}$ for d_{MRMS}) ensure that the obtained stego model is of very high visual quality and meanwhile prevent deforming too much the cover mesh. The geometric quality oriented protocol consists of the same steps; the difference is that we have different constraints on the induced objective and perceptual distortions as follows: $d_{MRMS} \leq 0.02\%.l_{bbd}$ and $d_{MSDM} \leq 0.30$. The constraint on d_{MRMS} guarantees that only a very small amount of geometric distortion is introduced to the cover mesh. The constraint on d_{MSDM} avoids this small-amount distortion (sometimes of high frequency) from degrading too much the visual quality of the deformed object. We are prepared to adjust these four thresholds according to the feedbacks from the research community. Finally, note that the two MSDM distance thresholds in the protocols correspond to the calculation in which the radius parameter is fixed as 0.005 [LGD*06]. This radius parameter is used to define the local window size during the calculation of the local MSDM distance as defined in Equation (8.3).

Both detectable and readable watermarking schemes can be tested by using our protocols. For readable schemes, we suggest to repeat the watermark embedding for at

least 5 times on each model and report the averages of the watermark extraction bit error rates (BER) under the different attacks. For detectable schemes, it is suggested that for each test model we repeat the watermark embedding for at least 100 times (instead of 5 times for readable schemes) by using different watermark sequences and keys. The receiver operating characteristics (ROC) curves (c.f. Section 5.5.3) under each kind of attacks are plotted as the evaluation results.

As pointed out in Chapter 3, robust mesh watermarking is a challenging task due to many particular difficulties and the relevant research is still in its early stage. We have taken into account this point when proposing the evaluation protocols, which are actually much less stringent compared to the protocols for image watermarking evaluation. First, it is acceptable that a readable mesh watermarking scheme has a relatively low capacity. Indeed, the amount of capacity depends heavily on the application. Low-capacity schemes can also be very useful, for example in the application of copy control examination in which a capacity of 2 bits (respectively designates "no right to copy", "the right to copy only once" and "the right to copy multiple times") seems already sufficient. We propose to set the capacity to one of the following values: 16 bits, 32 bits, 64 bits and $\geq$ 96 bits. However, when carrying out the comparison between different schemes, we always have to ensure that they have a same capacity. Second, instead of message error rate, we adopt the bit error rate as the robustness evaluation metric for readable mesh watermarking algorithms. If the message error rate were used, the decoding process would only output 0 (failure) or 1 (success) and a multi-bit message would be considered successfully decoded only if all the bits were correctly retrieved. We think that this evaluation metric is too stringent considering the state of the art in mesh watermarking and that it is more appropriate to use the bit error rate as the metric.

Finally, concerning the dataset collection, we have selected several representative meshes (with different vertex numbers and different shape complexities) as the test models, and also acquired the permission (from the Stanford Computer Graphics Laboratory and the Cyberware Inc.) to post them on our public server. These models are: Bunny (34835 vertices), Venus (100759 vertices), Horse (112642 vertices), Dragon (50000 vertices) and Rabbit (70658 vertices).

8.6 Comparison Results of Some Robust Algorithms

In order to test the utility of the proposed benchmarking software tool and protocols, we have used them to evaluate and compare several recent blind and robust readable mesh watermarking schemes: our wavelet-based method (presented in Chapter 5), our

Table 8.2: Baseline evaluation results of the first group tests (on the Venus model, with a capacity of 64 bits).

Methodology	Perceptual oriented protocol			Geometric oriented protocol		
Method	Wavelet	Cho's	Moment	Wavelet	Cho's	Moment
WM capacity (bits)	64	64	64	64	64	64
Embedding time (s)	12.8	7.6	439.9	12.6	11.6	377.6
Extraction time (s)	4.9	< 1.0	3.3	4.7	< 1.0	3.5
d_{MRMS} (w.r.t. l_{bbd})	0.078%	0.0080%	0.069%	0.019%	0.012%	0.018%
d_{MSDM}	0.10	0.19	0.14	0.05	0.29	0.09

moment-based method (presented in Chapter 6), our spectral-domain-based method (presented in Chapter 7), and the histogram-based method of Cho et al. [CPJ07]. Two groups of tests were carried out: in the first group, we tested Cho's method, the moment-based method and the wavelet-based method with a watermarking capacity of 64 bits, on the Venus model; in the second group, we tested Cho's method, the moment-based method and the spectral-domain-based method with a watermarking capacity of 16 bits, on the Rabbit model.

Results and analysis of the first group tests

Table 8.2 presents the baseline evaluation results of the first group tests (on the Venus model, with a capacity of 64 bits). The robustness evaluation results are presented in Table 8.3. All the results are the averages of 5 trials with randomly selected watermark sequences and keys. In order to apply the wavelet-based watermarking method, the original irregular Venus model is remeshed prior to the watermark embedding.

From these results, we can conclude that, for the Venus model, our moment-based method is more suitable to be used in applications that require a high visual quality of the watermarked object, while the method of Cho et al. is more appropriate for the applications which have strict restriction on the induced objective distortion. However, in both kinds of applications, if a strong robustness against connectivity attacks is required, then the moment-based method seems a better choice. The wavelet-based method show satisfactory and roughly comparable robustness performances (against geometry attacks) under both evaluation protocols. It is somewhat surprising that the robustness is not significantly deteriorated under the geometric quality oriented protocol test, considering that the watermark induced distortion under this protocol has been significantly decreased. For the wavelet-based method, we do not provide its robustness evaluation results against connectivity attacks since these attacks in general destroy the semi-regular connectivity of the stego mesh and thus makes it impossible to perform wavelet decomposition on the attacked models. Although in theory, the subdivision

Table 8.3: Robustness comparison of the first group tests (on the Venus model, with a capacity of 64 bits).

Methodology ⇒	Perceptual oriented protocol			Geometric oriented protocol		
Attack ⇓	Wavelet BER	Cho's BER	Moment BER	Wavelet BER	Cho's BER	Moment BER
File attack	0	0	0	0	0	0
Similarity transformation	0	0	0	0	0	0
Noise $A = 0.05\%$	0.03	0.01	0	0.04	0	0.02
Noise $A = 0.10\%$	0.06	0.03	0.01	0.12	0.01	0.15
Noise $A = 0.30\%$	0.20	0.13	0.08	0.39	0.10	0.29
Noise $A = 0.50\%$	0.29	0.28	0.16	0.55	0.24	0.40
Smoothing $N_{itr} = 5$	0.06	0.10	0	0.05	0.06	0.06
Smoothing $N_{itr} = 10$	0.06	0.23	0.01	0.06	0.16	0.18
Smoothing $N_{itr} = 30$	0.09	0.38	0.07	0.16	0.34	0.39
Smoothing $N_{itr} = 50$	0.17	0.45	0.14	0.24	0.42	0.51
Quantization $R = 11$	0.04	0	0	0.08	0	0.01
Quantization $R = 10$	0.03	0.04	0.01	0.06	0.02	0.17
Quantization $R = 9$	0.17	0.14	0.01	0.22	0.06	0.27
Quantization $R = 8$	0.27	0.26	0.05	0.45	0.18	0.39
Quantization $R = 7$	0.45	0.46	0.17	0.47	0.41	0.53
Subdivision Midpoint	N.A.	0.04	0	N.A.	0.02	0
Subdivision $\sqrt{3}$	N.A.	0.14	0	N.A.	0.09	0.01
Subdivision Loop	N.A.	0.16	0	N.A.	0.09	0.01
Simplification $E_{sim} = 10\%$	N.A.	0.01	0	N.A.	0	0
Simplification $E_{sim} = 30\%$	N.A.	0.05	0	N.A.	0.03	0
Simplification $E_{sim} = 50\%$	N.A.	0.18	0	N.A.	0.07	0.02
Simplification $E_{sim} = 70\%$	N.A.	0.33	0	N.A.	0.14	0.02
Simplification $E_{sim} = 90\%$	N.A.	0.23	0.01	N.A.	0.12	0.08
Simplification $E_{sim} = 95\%$	N.A.	0.38	0.01	N.A.	0.27	0.17
Simplification $E_{sim} = 97.5\%$	N.A.	0.47	0.05	N.A.	0.42	0.32
Cropping $V_{cr} = 10\%$	N.A.	0.50	0.51	N.A.	0.50	0.51
Cropping $V_{cr} = 30\%$	N.A.	0.53	0.49	N.A.	0.51	0.48
Cropping $V_{cr} = 50\%$	N.A.	0.51	0.49	N.A.	0.52	0.49

Table 8.4: Baseline evaluation results of the second group tests (on the Rabbit model, with a capacity of 16 bits).

Methodology	Perceptual oriented protocol			Geometric oriented protocol		
Method	Spectral	Cho's	Moment	Spectral	Cho's	Moment
WM capacity (bits)	16	16	16	16	16	16
Embedding time (s)	311.5	2.7	147.2	459.4	1.5	132.0
Extraction time (s)	47.6	< 1.0	3.0	49.0	< 1.0	3.0
d_{MRMS} (w.r.t. l_{bbd})	0.064%	0.058%	0.067%	0.017%	0.020%	0.018%
d_{MSDM}	0.09	0.19	0.12	0.06	0.08	0.09

attack keeps the semi-regular connectivity, but due to some implementation issues, we cannot perform wavelet decomposition on these extremely dense meshes because of the "out of memory" problem. One solution would be implementing an out-of-core mesh wavelet decomposition tool. However, we believe that our wavelet-based watermarking method possesses a high-level robustness against subdivision attacks because the watermark is embedded in the coarsest resolution level and the subdivision operation should not seriously affect the information at this level. Finally, all the three methods are fragile to cropping, which constitutes a very difficult attack to blind mesh watermarks.

In all, the advantage of the method of Cho et al. is that with a very low objective distortion induced by the watermark embedding, it can however resist very strong-amplitude attacks, and the main strengths of the moment-based method are its strong robustness against connectivity attacks and its high watermark imperceptibility. The wavelet-based method also has the advantage of being more imperceptible. In addition, it seems that the robustness performance of this method is not quite dependent on the embedding strength (especially under small and moderate amplitude attacks). This implies that the selected watermarking primitive, which can be equivalently considered as the ratio between the norm of a wavelet coefficient vector and the average length of the edges in the coarsest-level resolution (c.f. Section 5.2), may be a robust geometric feature of the cover mesh.

Results and analysis of the second group tests

Table 8.4 presents the baseline evaluation results of the second group tests (on the Rabbit model, with a capacity of 16 bits). The corresponding robustness evaluation results are presented in Table 8.5.

Under both protocols, our moment-based method seems the best choice. Our method is particularly robust against simplification and subdivision: we can always correctly extract the embedded watermark under these attacks, without any bit error. The method of Cho et al. has a very good performance under geometry attacks. However, com-

Table 8.5: Robustness comparison of the second group tests (on the Rabbit model, with a capacity of 16 bits).

Methodology ⇒	Perceptual oriented protocol			Geometric oriented protocol		
Attack ⇓	Spectral BER	Cho's BER	Moment BER	Spectral BER	Cho's BER	Moment BER
File attack	0	0	0	0	0	0
Similarity transformation	0	0	0	0	0	0
Noise $A = 0.05\%$	0	0	0	0.06	0	0
Noise $A = 0.10\%$	0.04	0	0	0.35	0	0
Noise $A = 0.30\%$	0.33	0.04	0	0.44	0.06	0.06
Noise $A = 0.50\%$	0.50	0.17	0.02	0.40	0.10	0.29
Smoothing $N_{itr} = 5$	0.02	0	0	0.40	0	0.06
Smoothing $N_{itr} = 10$	0.04	0	0	0.52	0.02	0.04
Smoothing $N_{itr} = 30$	0.33	0.19	0.04	0.50	0.08	0.40
Smoothing $N_{itr} = 50$	0.50	0.25	0.15	0.75	0.12	0.35
Quantization $R = 11$	0	0	0	0.19	0	0
Quantization $R = 10$	0.06	0	0	0.46	0.04	0
Quantization $R = 9$	0.29	0	0	0.35	0.04	0
Quantization $R = 8$	0.50	0.10	0	0.38	0.10	0.06
Quantization $R = 7$	0.48	0.28	0.06	0.38	0.21	0.56
Subdivision Midpoint	N.A.	0	0	N.A.	0	0
Subdivision $\sqrt{3}$	N.A.	0	0	N.A.	0	0
Subdivision Loop	N.A.	0	0	N.A.	0.02	0
Simplification $E_{sim} = 10\%$	0.02	0	0	0.04	0.02	0
Simplification $E_{sim} = 30\%$	0	0.12	0	0.04	0.17	0
Simplification $E_{sim} = 50\%$	0	0.19	0	0.15	0.33	0
Simplification $E_{sim} = 70\%$	0.08	0.25	0	0.17	0.41	0
Simplification $E_{sim} = 90\%$	0.17	0.23	0	0.35	0.31	0
Simplification $E_{sim} = 95\%$	0.40	0.23	0	0.58	0.36	0
Cropping $V_{cr} = 10\%$	0.47	0.47	0.45	0.56	0.32	0.54
Cropping $V_{cr} = 30\%$	0.49	0.48	0.49	0.51	0.44	0.46
Cropping $V_{cr} = 50\%$	0.50	0.47	0.43	0.48	0.42	0.51

pared to our moment-based method, their method is less robust against simplification, which constitutes the most realistic attack in practical applications. Our spectral method possesses a relatively satisfactory robustness under the perceptual quality oriented protocol (especially against simplification). Under the geometric quality oriented protocol, it has a very poor performance against geometry attacks, which means that this method cannot be used in computer-aided design and medical imaging applications. For the spectral method, the robustness evaluation results against subdivision attacks are missing because we encountered some implementation issues which lead to a memory crash when decomposing these very dense meshes. A stronger robustness can be attained if we increase the watermark embedding strength for the spectral method. As shown in Section 7.4, Tables 7.2 and 7.3, a much better robustness is achieved if the watermark induced distortion attains 2.37×10^{-3}, i.e. 0.13%.l_{bbd}. Indeed, this is a common problem for the mesh watermarking schemes that are based on the manifold harmonics transform because it is difficult to precisely control the amount of induced distortion due to the causality problem (c.f. Section 7.3). In fact, our method has already a much lower induced distortion than the method of Liu et al. [LPG08] (c.f. Tables 7.4 and 7.5). In the future, we would like to further decrease this watermark induced distortion while keeping the robustness performance, probably through the derivation of an elegant solution to the causality problem.

8.7 Conclusion

In this chapter, we proposed a benchmark for the evaluation of robust mesh watermarking schemes. MRMS is used to measure the objective distortion induced by the watermark embedding, while the perceptual distortion is evaluated by MSDM. A software tool including these two distortion metrics, as well as a large number of attacks, is implemented and made publicly available. Two mesh watermarking evaluation protocols are established: the perceptual quality oriented protocol is designed for the applications which require a high visual quality of the watermarked model and the geometric quality oriented protocol is to be used in the applications which have strict restriction on the induced objective distortion. Some recent blind mesh watermarking algorithms were compared within the proposed benchmarking framework. The data set, the protocol configuration file and the source code of the software are publicly available at `http://liris.cnrs.fr/meshbenchmark/`. We expect that the mesh watermarking researchers will contribute to this benchmarking work by providing new mesh models, new attacks, and most importantly the test results of their watermarking methods.

Chapter 9

Conclusion

9.1 Summary of Contributions

In this dissertation, we have presented our research study on digital watermarking of 3-D polygonal meshes. Our main objective was to construct several effective blind mesh watermarking methods. Indeed, the three different kinds of mesh watermarks (robust, fragile and high-capacity) have respectively promising applications (copyright protection, content authentication and content enhancement). This main objective was accomplished through the derivation of some quantization-based watermarking schemes in different mesh domains. The experimental results have demonstrated the effectiveness of the proposed blind schemes in terms of different performance metrics. The secondary objective of this thesis was to provide a benchmarking tool for robust mesh watermarking, so as to facilitate the evaluation and comparison of different algorithms. This secondary objective was achieved and the implemented open-source benchmarking system has been made freely accessible on the Internet.

The contributions of this dissertation work can be summarized as follows.

Comprehensive survey on mesh watermarking and attack-centric investigation
When carrying out the literature review, we first presented the existing mesh watermarking algorithms by classifying them as robust, fragile and high-capacity techniques. A requirement list was defined for each kind of techniques. After this routine presentation of the state of the art, an attack-centric investigation was provided. The attacks on watermarked meshes were classified; the existing counter-measures to resisting each kind of attacks were analyzed and discussed. The motivation of providing this attack-centric investigation was that the attacks play a very important role during the design

of a practical mesh watermarking algorithm. In fact, one of the specific difficulties for mesh watermarking, compared with image, audio and video watermarking, is the existence of many particular and intractable attacks. We hope that this new standing point of the literature review on mesh watermarking would be helpful to better understand the encountered difficulties and to discover promising future working directions.

Introducing scalar Costa quantization to watermarking 3-D meshes

Scalar Costa scheme is a quantization-based data embedding technique that has been widely used in blind watermarking of image, audio and video. The main advantages of this technique are its easy implementation and its high flexibility between capacity, distortion, robustness and security. In this thesis, we have introduced the scalar Costa scheme to watermarking 3-D meshes. In the volume-moment-based scheme, we have also slightly modified the original SCS in order to make it more adaptable to the specific watermark embedding space. By using the SCS, we successfully embedded multi-bit blind watermarks in three different mesh domains: the wavelet domain of a semi-regular mesh, and the spatial and spectral domains of a general mesh. The watermarking primitives, which are subject to scalar quantization, are respectively the norms and orientations of the wavelet coefficient vectors, the mesh local volume moments and the mesh manifold harmonics spectral amplitudes.

Multiple watermarking of semi-regular meshes

Sometimes it is necessary to embed a number of different watermarks in a same multimedia content for being used in different applications. For semi-regular meshes, we proposed a hierarchical multiple watermarking system based on the wavelet transform. Three different watermarks (robust, high-capacity and fragile) are embedded in different and appropriate resolution levels of a same semi-regular mesh. To the best of our knowledge, this hierarchical watermarking framework constitutes the first attempt on multiple watermarking of 3-D meshes in the literature.

Robust and blind watermarking based on 3-D shape descriptor

The analytic and continuous geometric volume moment is an intrinsic 3-D shape descriptor. This descriptor reflects the intrinsic property of the 3-D shape represented by the mesh model and is proven to be very robust against various attacks. We have proposed a robust and blind spatial watermarking technique for general meshes by using the volume moment as the watermarking primitive. The main strengths of this method are its high imperceptibility and its strong robustness against connectivity attacks. As far as we know, our method is the first mesh watermarking algorithm using a continu-

ous 3-D shape descriptor as the watermarking primitive, and it also constitutes the first attempt in the literature on achieving the robustness against 3-D shape representation conversions (discretization of the mesh into voxels).

Robust and blind spectral mesh watermarking
We have also proposed a robust and blind spectral mesh watermarking technique that makes use of the recently proposed manifold harmonics transform. The mesh spectrum coefficient amplitudes obtained by using this transform are quite robust against various attacks, including connectivity changes. A multi-bit watermark is embedded through an iterative quantization of part of the low-frequency manifold harmonics spectral amplitudes of the cover mesh. Our watermarking method is one of the few blind spectral mesh watermarking schemes that are computationally efficient and meanwhile robust against the connectivity attacks.

A publicly available benchmark for robust mesh watermarking
Finally, we have designed and implemented a benchmark for the evaluation of robust mesh watermarking methods. The proposed benchmark includes a "standard" mesh data set, a software tool and two performance evaluation protocols. The benchmark, which is an open source project, has been made publicly available at `http://liris.cnrs.fr/meshbenchmark/`. As far as we know, this is the first mesh watermarking benchmark implemented in the literature.

In all, an important characteristic of this dissertation work is that we have devised effective blind mesh watermarking schemes by combining useful ingredients from several different research domains. These ingredients include the scalar Costa scheme from the digital watermarking research, the volume moment from the shape analysis research, and the manifold harmonics transform from the geometry processing research. This interdisciplinary research method may also be considered as a contribution. Indeed, we believe that the future research on 3-D mesh watermarking is highly related to the advances in geometry processing and shape analysis (as can be observed in the next section), and that many exciting works will be realized if the experts from those different areas have consistent collaborations between them.

9.2 Perspectives

Concerning the perspectives, we distinguish between the short-term future work and the long-term working directions. The former consists of the improvements of the existing

works presented in different chapters of this manuscript.

Improvement of the wavelet-based multiple watermarking system
For the robust watermark, we would like to combine our scheme with advanced error correction coding methods (instead of the simple bit repetition), in order to enhance its robustness and/or to increase its capacity. For the high-capacity watermark, we intend to design an error correction code for the adopted permutation coding so as to make it less fragile. We also plan to try the idea of using local mesh geometric properties to synchronize the robust and high-capacity watermarks.

Improvement of the moment-based watermarking method
First of all, it is necessary to investigate the reason for the high-amplitude deformations of certain patches and afterward carry out some rectification of the watermarking algorithm so as to resolve this problem. An adaptable and robust mesh decomposition mechanism that produces patches with similar sizes is of our interest since it may allow to embed more bits in the cover mesh without degrading the watermark imperceptibility and robustness; this "intelligent" patch decomposition may also be helpful to solve the desynchronization problem caused by the patch classification.

Improvement of the spectral-domain-based watermarking method
First, it would be interesting to establish a theoretical explanation for the robustness of the manifold harmonics spectral coefficients. As mentioned in Section 7.5, our method may fail for certain objects due to the spectral coefficient robustness issue and the causality problem. Thus, it seems necessary to perform a comprehensive experimental study to better understand the behavior of the spectral coefficients under various attacks as well as the modulations due to the watermark embedding. For instance, a better robustness and a reduced watermark insertion time can be achieved if we prevent the bit embedding in the spectral coefficients which are intrinsically less stable. Ideally, a much better overall performance can be attained if we find an efficient and elegant way to solve the causality problem.

Improvement of the benchmarking system
We plan to continue this benchmarking work by providing more test meshes and attacks. In fact, it will be interesting to integrate the so-called estimation-based [VPP*01, PVM*01] attacks in the benchmark and also test the resistance of the recent mesh watermarking algorithms against these more effective attacks. Meanwhile, we are looking forward to receiving feedbacks from the research community, based on which we can improve the proposed benchmark and thus enhance its usability.

There still exist several open problems in the field of 3-D mesh watermarking research. The studies on these problems constitute our long-term future work.

Employing other 3-D shape descriptors for robust and blind mesh watermarking
It can be seen that 3-D shape descriptors can be very efficient mesh watermarking primitives. In fact, the histograms used in the methods of Zafeiriou et al. [ZTP05] and Cho et al. [CPJ07] are two statistical shape descriptors. The volume moment employed in our blind and robust scheme described in Chapter 6 is a transform-based shape descriptor. Therefore, it seems interesting to explore the possibility of using other 3-D shape descriptors for robust and blind mesh watermarking. Some examples of these promising descriptors are the 3-D Zernike moments [NK04], the 3-D angular radial transformation coefficients [RCB05] and the spherical harmonics transformation coefficients [FMK*03]. Some of them are particularly interesting because of their intrinsic invariance to rotation and their robustness against various operations. But unfortunately, the above descriptors are all defined on discretized voxel-based shape representations. Hence, it seems that we have to first of all derive their counterpart descriptors for 3-D meshes and then devise efficient watermarking algorithms by taking these derived descriptors as primitives.

Robustness against cropping combined with connectivity changes
This operation has been considered as the most intractable attack to a robust and blind mesh watermark. It seems that there exist two possible solutions: the first is to introduce a robust and "blind" mesh segmentation preprocessing that is capable of resisting this attack, and then repetitively embed the watermark in each segmented mesh patch; the second is to use a robust local shape descriptor as the watermarking primitive, which can still be successfully retrieved under cropping combined with connectivity alteration. When performing research on these two solutions, the mesh watermarking community may benefit from the recent achievements in the research on 3-D shape analysis and indexing, such as the work of Shapira et al. [SSCO08] and Liu et al. [LZSCO09].

Adaptive mesh watermarking
The performance of a mesh watermarking method can be improved if it takes the mesh local properties into account. For example, in the areas with low vertex sampling density or with high roughness, we can enhance the embedding strength of a robust scheme or increase the number of embedded bits for a high-capacity scheme.

High-capacity mesh watermark with invariance to all content-preserving operations
In many applications, we require that a high-capacity mesh watermark should be invariant to both element reordering and similarity transformation while providing a very

high payload. In order to achieve this target, the wavelet-based high-capacity scheme presented in Chapter 5 applies the basic idea of permutation steganography when embedding a watermark in the mesh geometry. It would be interesting to investigate whether it is possible to devise a similar scheme for arbitrary meshes.

Fully functional fragile watermark for arbitrary meshes
Although in Chapter 5 we devised an effective fragile watermarking method for semi-regular meshes, it seems much more difficult to design such a scheme for arbitrary meshes. The main difficulty consists in how to achieve the following properties at the same time: the immunity to causality problem, the invariance to all the content-preserving operations, a high-level security and the numerical stability.

Deformation-invariant watermarking
A 3-D mesh may be subject to some realistic (and high-amplitude) deformations to form a sequence of moving objects. For instance, we can make the Horse model run, or make the Venus model have different facial expressions. Ideally, a robust watermark should be able to resist these realistic deformations of the stego model. But unfortunately, it seems very difficult to build such a deformation-invariant blind mesh watermark. One possible solution would be looking for an invariant 3-D shape descriptor and using it as the watermarking primitive. Once again, we may benefit from the achievements in the shape analysis research, especially from the studies on deformation-invariant shape representation [EK03, JZ07, Rus07].

Some other open questions are: What is the "best" mesh spectral transform with regard to the robust and blind mesh watermarking? How and where can we add a watermarking functionality in a mesh compression chain? Is it possible to protect progressive meshes by means of watermarking? Is it possible to use watermarking to realize mesh or mesh sequence compression? Finding answers to these questions also constitute several interesting future working directions.

In all, 3-D mesh watermarking is a challenging and interesting research area, with many open problems and potential applications. We believe that the study on this research area has a very promising future, especially with the joint effort of the experts from the different research communities such as digital watermarking, shape analysis and geometry processing.

Appendix A

Résumé en Français

Sommaire

Cette annexe est un résumé long en français de ce manuscrit de thèse. Après une introduction du contexte et des objectifs de ce travail qui concerne le tatouage numérique de maillages tridimensionnels (3D), nous présentons un bref état de l'art du domaine. Puis, les différentes méthodes aveugles développées sont présentées ; elles reposent sur la quantification scalaire de Costa et utilisent différents domaines du maillage. Nous présentons ensuite notre travail sur le benchmarking des techniques de tatouage robustes de maillages 3D. Enfin, les contributions de cette thèse sont résumées et plusieurs directions de recherche et perspectives sont présentées.

A.1 Introduction

A.1.1 Le tatouage de maillages 3D et ses applications

Avec l'amélioration de la puissance des ordinateurs personnels et l'augmentation de la vitesse de transmission des réseaux, les modèles tridimensionnels (3D) sont de plus en plus utilisés dans différentes applications telles que l'imagerie médicale, la simulation scientifique, les jeux vidéos et la conception assistée par ordinateur. Un modèle 3D est souvent représenté numériquement par un maillage polygonal, c'est-à-dire un ensemble de facettes polygonales visant à constituer une bonne approximation de la surface de l'objet 3D (cf. Figure A.1). Un maillage polygonal contient deux types d'informations : l'information de *géométrie* qui représente les coordonnées 3D des sommets du maillage, et l'information de *connectivité* qui décrit la relation d'adjacence (les arêtes) entre les sommets. Bien qu'il existe de nombreuses autres représentations 3D (par exemple surfaces implicites, NURBS ou voxels), le maillage 3D est devenu *de facto* le standard de la représentation numérique des objets 3D grâce à sa simplicité algébrique. Il est considéré comme un modèle de bas niveau mais très efficace. En outre, il est relativement facile de convertir d'autres représentations en maillages 3D

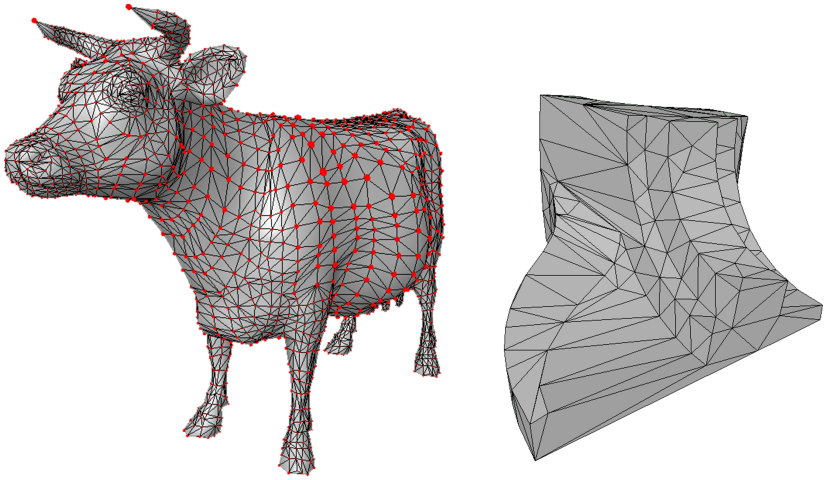


FIGURE A.1 – Deux exemples de maillages 3D : la Vache (à gauche) et le Fandisk (à droite).

Malheureusement, comme les images, les vidéos et les fichiers audio numériques, les maillages 3D peuvent être facilement reproduits et redistribués par un pirate, sans aucune perte de qualité. Ce comportement illégal porte préjudice au droit d'auteur des propriétaires des maillages et peut également nuire au fonctionnement des chaînes commerciales qui sont basées sur ces maillages. En effet, la génération des maillages 3D, soit par numérisation des objets 3D réels ou en utilisant un logiciel de conception spécifique,

est habituellement un processus très coûteux. La technique du tatouage robuste apparaît être une bonne solution à ce problème de protection de la propriété intellectuelle des maillages 3D. Cette technique consiste à cacher une certaine quantité d'information secrète (c.-à-d. le tatouage) dans la partie utile du fichier à protéger. Le tatouage inséré doit être *robuste* aux opérations ordinaires ainsi qu'aux attaques malveillantes sur le modèle tatoué, et en même temps être visuellement *imperceptible*. Le tatouage inséré peut être par exemple l'identifiant numérique de la compagnie qui détient la propriété du maillage ; ainsi, dans le cas de litiges, l'entreprise peut extraire le tatouage inséré pour justifier sa propriété légale sur le modèle. Il peut aussi s'agir d'un numéro de série permettant d'identifier l'origine de l'objet piraté. En plus du tatouage robuste utilisé pour la protection de la propriété intellectuelle, le tatouage *fragile* et le tatouage de *haute capacité* possèdent également de nombreuses applications potentielles telles que l'authentification du maillage et l'enrichissement du contenu. Un tatouage fragile est conçu pour être intentionnellement vulnérable à certaines attaques dites *non-tolérables*. Un échec à l'extraction de ce tatouage indique alors l'existence d'une attaque sur le modèle tatoué. Un tatouage de haute capacité est capable de transporter une très grande quantité d'information. En général, l'objectif d'un tel tatouage est simplement de cacher des informations auxiliaires (par exemple l'adresse d'une page web expliquant l'utilisation du fichier ou un tag d'indexation) dans le contenu original, afin de renforcer l'utilité du contenu ou de fournir un service supplémentaire.

Depuis les tout premiers travaux de Ohbuchi et coll. [OMA97] publiés en 1997, une attention croissante a été portée sur la recherche des techniques de tatouage pour les maillages 3D (cf. Figure A.2). Nous pouvons imaginer les scénarios d'application suivants de ces techniques.

- Un constructeur automobile peut insérer son identifiant numérique dans les pièces de voiture qu'il a conçues ; ce tatouage peut alors être extrait pour prouver les propriétés légales du constructeur sur ces pièces.
- L'acheteur d'un fichier de maillage 3D peut vérifier l'intégrité et authentifier l'origine du modèle qu'il a reçu à partir du résultat de l'extraction du tatouage fragile inséré. Si la méthode de tatouage fragile est bien conçue, elle permet aussi de localiser ou même de réparer les parties attaquées.
- Un médecin peut cacher les informations personnelles d'un patient dans le maillage 3D obtenu après un examen 3D de type scanner X ou IRM, sous forme d'un tatouage de haute capacité (à condition que cette insertion n'ait aucune conséquence sur le diagnostic), pour éviter de se tromper sur la correspondance entre les infor-

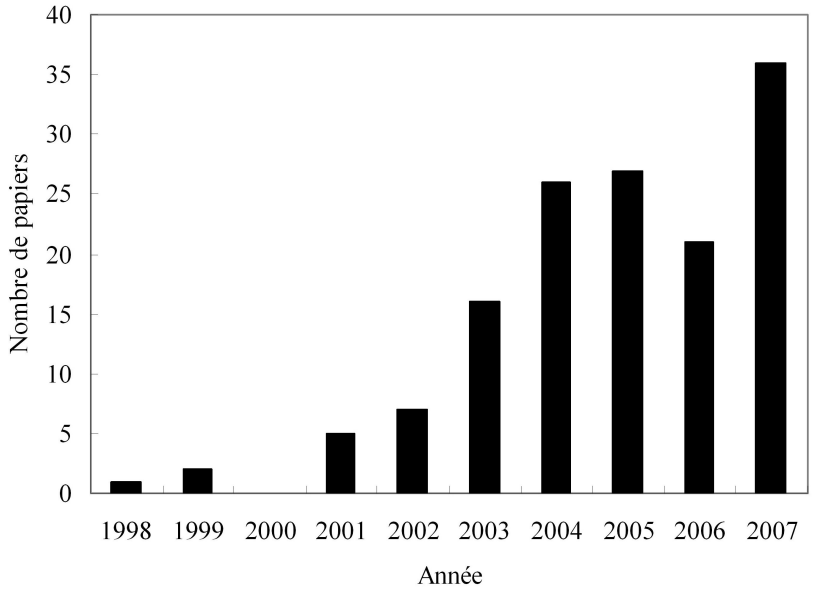


FIGURE A.2 – Nombre croissant des articles scientifiques sur le tatouage de maillages 3D qui sont indexés par EI Compendex.

mations personnelles du patient et son résultat d'examen.

- La texture d'un maillage ou même les paramètres d'animation d'une séquence de maillages pourraient être cachés dans le maillage en utilisant une méthode de tatouage de haute capacité pour réaliser la compression du contenu. Cette application est similaire à celle de la compression de vidéo où la partie audio du clip vidéo est cachée dans la partie visuelle via la technique de tatouage.

A.1.2 Objectifs et contributions

Le sujet de recherche de cette thèse est le tatouage numérique de maillages 3D. Notre objectif principal est de construire des méthodes de tatouage *aveugles*, qui ne nécessitent pas le maillage original non-tatoué pour l'extraction du tatouage. Nous nous concentrons sur la recherche de méthodes aveugles car elles possèdent une gamme d'applications beaucoup plus large que les méthodes non-aveugles. En effet, en contexte réel, il est normal que le contenu original ne puisse pas, ou même ne doive pas être présent lors de la phase d'extraction du tatouage, souvent pour des raisons d'efficacité de l'algorithme ou de sécurité. Par exemple, dans l'application de contrôle de copies, il est inapproprié de fournir la version d'origine au dispositif de contrôle qui est probablement dans la main d'un client malveillant. Il est aussi inutile de concevoir un tatouage fragile non-aveugle pour l'authentification du contenu, parce que la tâche d'authentification devient triviale, voire superflue si le récepteur possède la version originale. Concernant

la recherche sur le tatouage robuste et aveugle de maillages 3D, nous souhaitons élaborer des méthodes qui sont capables de résister aux attaques de *connectivité*. Alors que les attaques de *géométrie* modifient seulement les coordonnées des sommets du maillage tatoué, les attaques de connectivité modifient aussi les relations d'adjacence entre les sommets. Quelques exemples typiques d'attaques de connectivité incluent la simplification, la subdivision et le remaillage. Dans ces opérations, les sommets, arêtes et facettes d'origine peuvent être retirés du maillage tatoué ; en même temps, des nouveaux sommets, arêtes et facettes peuvent être y insérés. En pratique, il est très difficile d'élaborer une méthode de tatouage aveugle qui soit robuste à ces attaques de connectivité.

Afin d'atteindre l'objectif principal de cette thèse, nous utilisons le schéma de Costa scalaire (SCS) [EBTG03] pour construire des méthodes de tatouage aveugles. Le SCS est une technique de quantification largement utilisée pour le tatouage aveugle des fichiers image, audio et vidéo ; dans cette thèse, nous introduisons le SCS pour tatouer les maillages 3D dans différents domaines pertinents. Les domaines exploités incluent le domaine d'ondelettes d'un maillage semi-régulier, et les domaines spatial et spectral d'un maillage manifold de connectivité quelconque. Les primitives de tatouage sont respectivement les coefficients d'ondelettes [LDW97], les moments volumiques analytiques [ZC01] et les amplitudes spectrales en harmoniques variétés [VL08]. Dans la méthode basée sur les ondelettes, trois tatouages aveugles différents (robuste, de haute capacité et fragile) peuvent être insérés dans un même maillage semi-régulier sans aucune interférence entre eux. La méthode basée sur les moments et la méthode basée sur l'analyse spectrale possèdent une très bonne imperceptibilité ainsi qu'une très forte robustesse aux attaques de connectivité.

Parallèlement à l'élaboration des techniques de tatouage aveugles, nous souhaitons également fournir à la communauté un système de benchmark pour les méthodes de tatouage robustes de maillages polygonaux. L'objectif de la construction d'un tel système est de faciliter l'évaluation et la comparaison des différentes méthodes. Ce benchmark a été implémenté et rendu accessible sur Internet à l'adresse suivante : `http://liris.cnrs.fr/meshbenchmark/`.

A.1.3 Organisation du résumé

Le reste du résumé est organisé de la manière suivante :

La section A.2 présente quelques connaissances de base sur les maillages 3D et sur le tatouage numérique nécessaires pour comprendre les sections suivantes.

La section A.3 fournit une étude bibliographique sur le tatouage de maillages 3D.

La section A.4 introduit brièvement le schéma de Costa scalaire, qui sera utilisé comme technique de dissimulation de données par les méthodes de tatouage aveugles proposées dans les sections A.5-A.7.

La section A.5 propose un système de tatouage multiple et hiérarchique pour les maillages semi-réguliers, basé sur la transformation en ondelettes. Trois tatouages aveugles différents sont insérés à différents niveaux de résolution du maillage original, par une modification des coefficients d'ondelettes associés à ces niveaux.

La section A.6 décrit une technique de tatouage robuste et aveugle dans le domaine spatial d'un maillage manifold de connectivité quelconque. La robustesse de cette technique repose sur la stabilité des moments volumiques analytiques et continus du maillage.

La section A.7 présente une méthode robuste et aveugle dans le domaine spectral, à la fois efficace en termes de temps de calcul et robuste aux attaques de connectivité. Les primitives de tatouage sont les amplitudes spectrales de basse fréquence du maillage, obtenues par la transformation en harmoniques variétés.

La section A.8 décrit un système de benchmark pour les techniques de tatouage robustes de maillages 3D, qui comprend une collection «standard» de modèles maillés, un outil logiciel et deux protocoles d'évaluation.

La section A.9 résume les contributions de cette thèse et propose plusieurs directions de recherche prometteuses.

A.2 Connaissances de base

A.2.1 Maillage polygonal

Un maillage 3D contient trois éléments combinatoires différents : les *sommets*, les *arêtes* et les *facettes* (typiquement triangles ou quadrangles). Les coordonnées des sommets constituent l'information de *géométrie* du maillage, tandis que les arêtes et les facettes décrivent les relations d'adjacence entre les sommets et constituent l'information de *connectivité* du maillage. Mathématiquement, un maillage $\mathcal{M}$ qui contient N_V sommets et N_E arêtes peut être modélisé par un signal $\mathcal{M} = \{\mathcal{V}, \mathcal{E}\}$, où

$$\mathcal{V} = \{v_i = (x_i, y_i, z_i) \,|i \in \{1, 2, ..., N_V\}\}\,, \tag{A.1}$$

$$\mathcal{E} = \left\{e_j := \left(p_1^{(j)}, p_2^{(j)}\right) |j \in \{1, 2, ..., N_E\}\,; p_1^{(j)}, p_2^{(j)} \in \{1, 2, ..., N_V\}\right\}. \tag{A.2}$$

Plus précisément, chaque sommet v_i est décrit par ses coordonnées 3D (x_i, y_i, z_i) ; chaque élément dans $\mathcal{E}$ représente une arête reliant deux sommets indexés respectivement par $p_1^{(j)}$ et $p_2^{(j)}$. La *valence* d'un sommet est le nombre de ses arêtes incidentes, et le *degré* d'une facette est simplement le nombre de ses arêtes.

Au lieu de la liste des arêtes $\mathcal{E}$, la connectivité du maillage $\mathcal{M}$ peut être aussi complètement décrite par une liste de ses N_F facettes

$$\mathcal{F} = \left\{ f_k := \left(p_1^{(k)}, p_2^{(k)}, ..., p_D^{(k)} \right) | k \in \{1, 2, ..., N_F\} \right\}, \tag{A.3}$$

où D est le degré de la facette f_k et

$$p_d^{(k)} \in \{1, 2, ..., N_V\}, \left(p_{d-1}^{(k)}, p_d^{(k)} \right) \in \mathcal{E};\ d \in \{1, 2, ..., D\}, p_0^{(k)} := p_D^{(k)}. \tag{A.4}$$

Nous pouvons constater que chaque facette dans la liste $\mathcal{F}$ est représentée par une séquence des indices de ses sommets ordonnés cycliquement autour de la facette. Il existe un certain nombre de formats de stockage pour les maillages polygonaux, tels que le «3-D Object File Format (OFF)», le «wavefront OBJect format (OBJ)», le «Stanford University PoLYgon format (PLY)» et le «Virtual Reality Modeling Language (VRML)». Tous ces formats adoptent une stratégie similaire pour stocker un maillage d'une manière non-compressée, sous forme d'une liste de sommets suivie par une liste de facettes.

Un maillage est dit *triangulaire* si toutes ses facettes sont des triangles ; d'une manière similaire, nous pouvons définir un maillage *quadrangulaire*. Un maillage est qualifié de *régulier* si tous ses sommets ont une même valence et de *semi-régulier* s'il est régulier par morceau, donc s'il possède un grand nombre de sommets réguliers. Sinon, le maillage est dit *irrégulier*. Nous disons qu'un maillage est *manifold* si le voisinage de chaque sommet est homéomorphe à un disque ou un demi-disque. L'*orientation* d'une facette est définie en fonction de l'ordre cyclique de ses sommets combiné avec la règle de la main droite. Evidemment, il existe deux possibilités pour cette orientation. Les orientations des deux facettes adjacentes sont appelés *compatibles* si leurs deux sommets communs sont dans des ordres opposés à l'intérieur de ces deux facettes. Le maillage entier est appelé *orientable* si nous pouvons trouver une combinaison des orientations de toutes ses facettes telle que les orientations de chaque paire de facettes adjacentes soient compatibles.

A.2.2 Tatouage numérique

L'idée de base des techniques de tatouage numérique [KP00, BB04, CMB*07] est de cacher une information secrète (le tatouage ou la marque) dans la partie utile du contenu

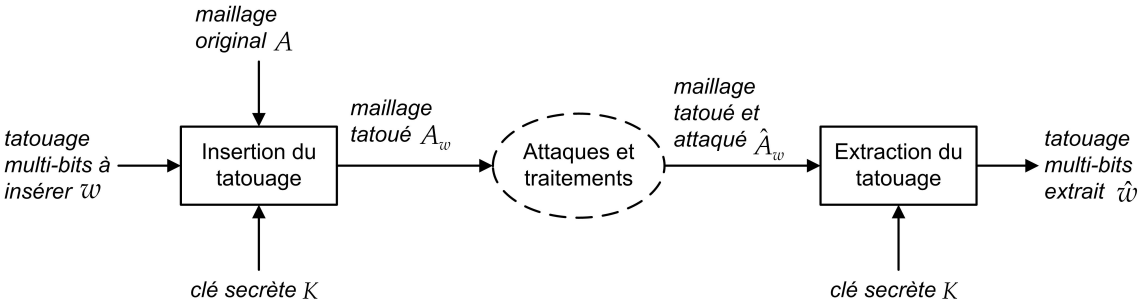


FIGURE A.3 – Schéma général des méthodes de tatouage aveugles et lisibles. Dans cette thèse, nous nous concentrons sur la recherche de ce type de méthodes.

multimédia à protéger. Dans la terminologie du tatouage, le contenu original dans lequel nous insérons une marque est souvent appelé le contenu *hôte*, tandis que le contenu tatoué est aussi appelé le contenu *stégo*. Par rapport à la cryptographie [MvOV01], le tatouage numérique est capable de protéger les œuvres digitales après la phase de transmission et l'accès légal, car le tatouage est mélangé avec la partie utile du fichier multimédia et donc coexiste toujours avec le contenu protégé.

Il existe différentes classifications pour les algorithmes de tatouage. Tout d'abord, nous distinguons les techniques *non-aveugles* et les techniques *aveugles*. Les méthodes non-aveugles ont besoin du contenu original non-tatoué pour extraire le tatouage ; par contre, les méthodes aveugles ne le nécessitent pas. Nous classifions également le tatouage comme étant *robuste*, *fragile* ou de *haute capacité*, selon les applications visées (respectivement la protection de propriété intellectuelle, l'authentification du fichier multimédia et l'enrichissement du contenu hôte). En pratique, nous distinguons également les algorithmes de tatouage *spatiaux/temporels* et les algorithmes basés sur une *transformation*, selon l'espace d'insertion du tatouage. Enfin, les techniques de tatouage peuvent être aussi classifiées comme étant *détectables* ou *lisibles*. Dans une méthode détectable, après la détection du tatouage, nous pouvons seulement dire si le contenu multimédia considéré a contient une marqué donnée ou non. Cela signifie que nous n'obtenons qu'une réponse binaire qui notifie l'existence d'un certain tatouage dans un certain fichier multimédia. Par contre, dans une méthode lisible, nous pouvons extraire et décoder un message composé de plusieurs bits transporté par le tatouage.

Dans cette thèse, nous nous concentrons sur la recherche de méthodes de tatouage aveugles et lisibles pour les maillages 3D. Le schéma général de ce type d'algorithmes est illustré dans la figure A.3. Nous nous intéressons principalement au tatouage robuste, mais aussi à l'élaboration des méthodes fragile et de haute capacité.

Un système de tatouage est souvent évalué un utilisant quatre critères différents : la

capacité, la *distorsion*, la *robustesse* et la *sécurité*. La capacité désigne le nombre de bits du message caché transporté par le tatouage. La distorsion mesure la différence (*objective* ou *perceptuelle*) entre le contenu original et le contenu tatoué. La robustesse indique la résistance du tatouage face aux différentes opérations ou attaques. Une méthode de tatouage sécurisée doit être capable de résister aux attaques très malveillantes qui visent à détruire le système d'authentification ou de protection de copyright basé sur le tatouage, au travers, par exemple, de la révélation de clé secrète ou de l'inversion de la procédure de l'insertion de tatouage. La sécurité est plutôt considérée comme une demande de haut niveau pour une méthode de tatouage.

Après avoir présenté dans cette section quelques connaissances de base sur les maillages 3D et sur le tatouage numérique, dans la section suivante, nous allons donner une vue globale sur l'état de l'art en tatouage de maillages 3D.

A.3 Etat de l'art en tatouage de maillages 3D

A.3.1 Difficultés et classification

Jusqu'à présent, il existe encore peu de méthodes de tatouage pour les maillages 3D, par rapport à la maturité relative de la recherche sur le tatouage des images, des fichiers audio ou des vidéos. Cette situation est principalement due à deux difficultés : 1) l'échantillonnage irrégulier des maillages, et 2) la complexité des attaques possibles sur les modèles tatoués. Ces deux problèmes sont expliqués dans les paragraphes suivants.

Dans le cas du tatouage des images 2D, l'image originale peut être considérée comme une matrice, et chaque pixel comme un élément de cette matrice. Cela signifie que tous les pixels ont un ordre intrinsèque dans l'image (par exemple l'ordre établi par le balayage en ligne ou en colonne). Cet ordre est souvent utilisé pour *synchroniser* le tatouage, c'est-à-dire pour savoir où les éléments du signal de tatouage w sont insérés, et dans quel ordre. Au contraire, il n'existe pas d'ordre simple, intrinsèque et robuste pour les éléments combinatoires d'un maillage, qui véhiculent les éléments du signal de tatouage (c.-à-d. les primitives de tatouage). Certains ordres intuitifs, comme l'ordre des sommets et des facettes dans le fichier du maillage ou l'ordre des sommets obtenus par le classement de leurs projections sur un axe du système de coordonnées objectives, sont faciles à modifier. En outre, en raison de l'échantillonnage irrégulier, il nous manque encore un outil efficace pour effectuer une analyse spectrale sur les maillages. Par conséquent, il est très difficile de construire une méthode de tatouage performante dans le domaine spectral.

En plus du point ci-dessus, les tatouages robustes doivent également affronter diverses attaques très difficiles. Le réarrangement des sommets et des facettes dans le fichier de données n'a aucun impact sur la forme du maillage tatoué, mais il peut désynchroniser les tatouages qui reposent sur cet ordre intuitif. Les transformations de similarité, telles que la translation, la rotation, la mise à l'échelle uniforme et leurs combinaisons, sont censées être des opérations ordinaires à laquelle un tatouage robuste, ou même un tatouage fragile ou de haute capacité, doit être capable de résister. Dans des attaques plus sévères telles qu'une simplification ou un remaillage du modèle tatoué, les primitives de tatouage originales (par exemple des sommets, arêtes ou facettes) peuvent tout simplement disparaître. Ces attaques peuvent être facilement exercées à l'aide de différents outils logiciels disponibles gratuitement sur Internet, comme ReMESH [AF06] et MeshLab [CCR08], et elles peuvent détruire complètement la géométrie et la connectivité d'un maillage tatoué tout en préservant sa forme globale. Comme mentionné dans la section A.1.2, en général nous distinguons les attaques de géométrie (par exemple l'ajout de bruit, le lissage et la quantification des coordonnées des sommets) des attaques de connectivité (par exemple la simplification, la subdivision, le remaillage et la coupe). La figure A.4 illustre le modèle Lapin original et quelques versions attaquées.

Comme précisé dans la section A.1.2, l'objectif principal de cette thèse est de construire des méthodes de tatouage aveugles et robustes pour les maillages 3D. Cependant, en raison des difficultés présentées ci-dessus, il semble difficile d'atteindre cet objectif. En effet, au niveau théorique, il a été prouvé que l'exigence d'être aveugle ne cause aucune perte de performance pour une méthode de tatouage, au moins sous certaines hypothèses [Cos83]. Mais en pratique, un tatouage aveugle est normalement beaucoup moins robuste qu'un tatouage non-aveugle. Dans le cas non aveugle, la connaissance du maillage original rend l'extraction (ou la détection) du tatouage beaucoup moins difficile, principalement dans le sens où elle peut faciliter le processus de synchronisation du tatouage, en particulier sous les attaques de connectivité. Par conséquent, les principales difficultés rencontrées lors de l'élaboration d'un tatouage aveugle et robuste pour les maillages 3D consistent à trouver une primitive de tatouage appropriée et à établir un mécanisme de synchronisation robuste.

Dans la suite, nous présenterons les algorithmes de tatouage de maillages existants en les classifiant comme fragiles, de haute capacité et robustes. Dans chaque classe, il semble convenable de diviser les membres en deux sous-classes, selon que le tatouage est inséré dans le domaine spatial ou dans un domaine transformé. Dans une méthode spatiale, le tatouage est inséré directement en modifiant la géométrie ou la connectivité du maillage, tandis que dans une méthode par transformée, le tatouage est inséré en

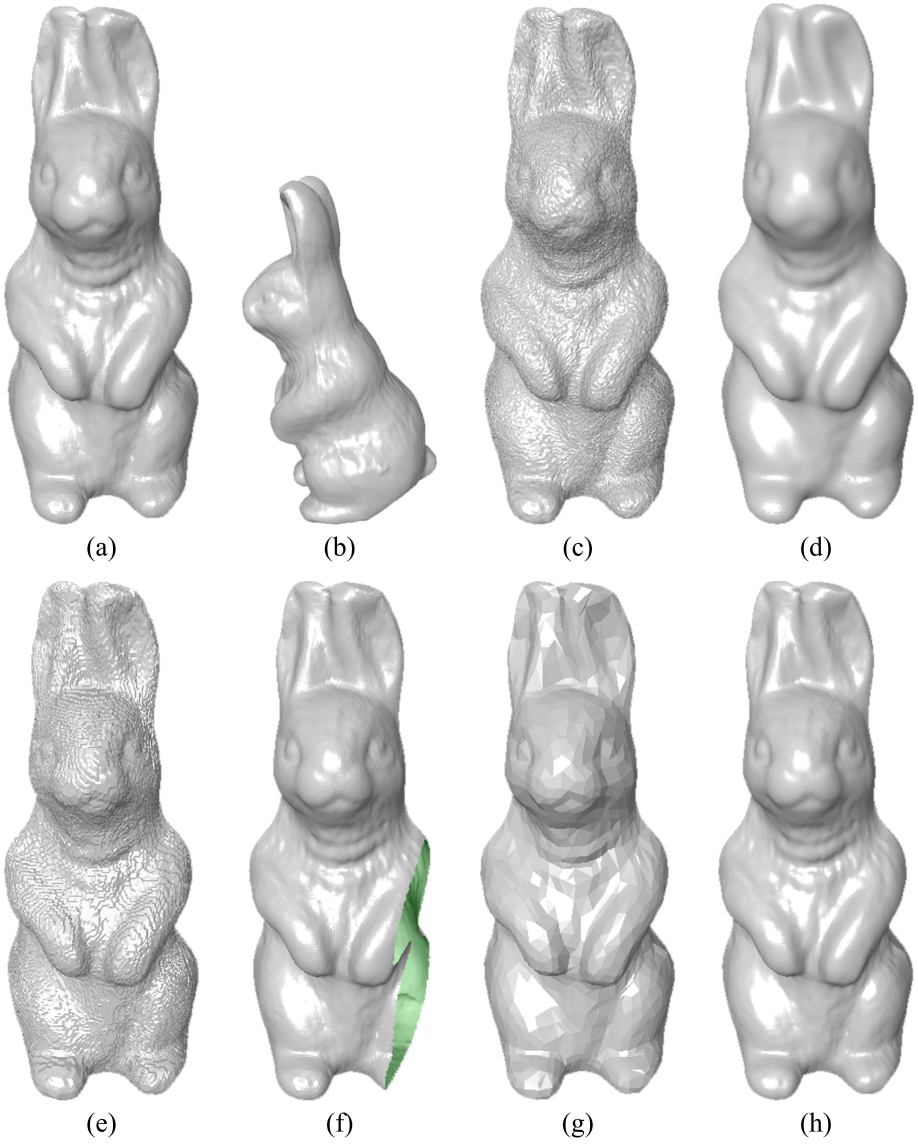


FIGURE A.4 – Le modèle Lapin original et quelques versions attaquées : (a) le modèle original ; (b) après une transformation de similarité ; (c) après un ajout de bruit pseudo-aléatoire ; (d) après un lissage ; (e) après une quantification des coordonnées des sommets ; (f) après une coupe ; (g) après une simplification ; (h) après une subdivision.

TABLE A.1 – Comparaison entre les différentes méthodes de tatouage fragiles.

Méthode	Invariance au réarrangement des éléments	Invariance à la transformation de similarité	Localisation des attaques
Yeo & Yeung [YY99]	Non	Non	Oui
Lin et coll. [LLLL05]	Oui	Non	Oui
Chou & Tseng [CT06]	Oui	Non	Oui
Wu & Chueng [WC06]	Non	Oui	Oui
Wang et coll. [WZYG08]	Oui	Non	Oui
Cho et coll. [CLLP05]	Oui	Oui	Pas précisément

modulant des coefficients obtenus après une certaine transformation du maillage original.

A.3.2 Méthodes fragiles

Une technique de tatouage fragile utilisable dans des applications d'authentification de maillages doit posséder deux propriétés : 1) elle doit être vulnérable aux modifications sur le maillage tatoué, même avec des amplitudes très faibles, et 2) l'extraction devrait être capable de localiser, voire d'identifier les attaques subies. Cependant, nous voulons souvent un tatouage (semi-)fragile pouvant être robuste aux opérations préservant le contenu (*opérations «content-preserving»*), qui incluent le réarrangement des sommet/facettes dans le fichier de données (également appelé le *réarrangement des éléments combinatoires*), et la transformation de similarité. Dans de nombreuses applications, ces opérations ne sont pas considérées comme des attaques malveillantes, mais plutôt comme des opérations ordinaires, car théoriquement elles n'ont aucune influence sur la forme du maillage. En même temps, un tatouage fragile de maillages doit être aveugle, car en général il est inutile de construire un algorithme d'authentification non-aveugle (cf. Section A.1.2).

Le tableau A.1 compare les différentes méthodes de tatouage fragiles proposées jusqu'à présent. Aucune de ces méthodes n'est invariante à la fois au réarrangement des éléments combinatoires et à la transformation de similarité, tout en possédant la capacité de localiser précisément les attaques subies.

A.3.3 Méthodes de haute capacité

L'objectif d'un tatouage de haute capacité est de dissimuler une grande quantité d'informations auxiliaires au sein du modèle hôte. Dans la plupart des applications, le tatouage inséré doit être invariant aux opérations «content-preserving». Une méthode de haute capacité doit aussi généralement être aveugle, afin de faciliter l'extraction du tatouage et donc d'augmenter le champ d'application de l'algorithme.

TABLE A.2 – Comparaison entre les différentes méthodes de haute capacité.

Catégorie	Algorithme	Capacité	Invariance au réarrangement des éléments	Invariance à transformation de similarité
Méthodes basées sur la géométrie	Cayre & Macq [CM03]	$\approx$ 1 bit/sommet	Oui	Oui
	Wang & Cheng [WC05]	$\approx$ 3 bits/sommet	Oui	Oui
	Cheng & Wang [CW07]	3 $\sim$ 6 bits/sommet	Oui	Oui
	Tsai et coll. [TWC*06]	3 bits/sommet	Oui	Oui
Méthodes basées sur l'order	Cheng & Wang [CW06]	$\approx$ 6 bits/sommet	Non	Oui
	Bogomjakov et coll. [BGI08]	$((\log_2 n!) - n + 1)^*$	Non	Oui

*Pour un ensemble de n éléments. Cet ensemble peut être les N_V sommets or les N_F facettes du maillage.

Les algorithmes de tatouage de haute capacité de maillages 3D peuvent être divisés en deux groupes : les méthodes basées sur la modification de la géométrie du maillage et les méthodes basées sur des changements de l'ordre des éléments combinatoires dans le fichier de données. Le tableau A.2 compare les différentes méthodes de haute capacité existantes, en termes de capacité et de robustesse aux opérations «content-preserving». Toutes les techniques présentées dans le tableau A.2 sont aveugles.

A.3.4 Méthodes robustes

Une technique robuste doit au moins être capable de résister aux attaques qui introduisent des distorsions inférieures à un certain seuil au delà duquel le maillage tatoué et attaqué est fortement dégradé. En même temps, nous devons également nous assurer que la distorsion objective et/ou perceptuelle introduite par l'insertion du tatouage est sous la tolérance de l'application visée. En ce qui concerne la capacité du tatouage, pour les méthodes détectables utilisées dans l'application de vérification des droits d'auteur, seulement 1 bit d'information doit être inséré (normalement sous forme d'une séquence de nombres pseudo-aléatoires). Pour les méthodes lisibles, la capacité dépend fortement de l'application visée : par exemple, dans l'application de contrôle de copie, une faible capacité de seulement quelques bits est suffisante ; par contre, dans l'application typique de protection de la propriété intellectuelle, nous devons en général assurer une capacité d'environ 70 bits, qui permet d'insérer les identifiants numériques du propriétaire du contenu, de l'acheteur et du contenu lui-même [KP99].

Les tableaux A.3 et A.4 présentent les comparaisons entre les différentes méthodes robustes existantes. Les valeurs dans la colonne «Nombre de bits insérés» sont celles rapportées dans les papiers originaux. La robustesse à la plupart des attaques est évaluée d'une manière qualitative par un signe allant de «$--$», qui signifie le moins robuste, à «$++$», qui signifie le plus robuste. Dans ces deux tableaux, les algorithmes robustes sont classifiés selon le domaine d'insertion et la primitive de tatouage. Les catégories

TABLE A.3 – Comparaisons entre les différentes méthodes robustes.

Catégories	Algorithmes	Nombre de bits insérés	Aveugle ?	Adaptabilité locale
Techniques spatiales sur sommets	Yu et coll. [YIK03]	$\approx$ 50 bits	Non	Oui
	VFA [Ben99b]	$\approx$ 900 bits	Oui	Non
	Zafeiriou et coll. [ZTP05]	$\approx$ 20 bits	Oui	Non
	Cho et coll. [CPJ07]	64 bits	Oui	Non
	Bors [Bor06]	$\approx$ 0.2 bits/sommet	Oui	Oui
Techniques spatiales sur facettes	TSQ [OMA97]	$\approx$ 1.2 bits/facette	Oui	Non
	Benedens [Ben99a]	$\approx$ 30 bits	Semi	Non
	Lee et coll. [LK07]	$\approx$ 50 bits	Semi	Oui
Techniques basées sur analyse spectrale directe	Ohbuchi et coll. [OMT02]	32 bits	Non	Non
	Cayre et coll. [CRAS*03]	64 bits	Oui	Non
	Wu & Kobbelt [WK05]	24 bits	Non	Non
	Rondao-Alface & Macq [RAM05]	64 bits	Oui	Non
	Liu et coll. [LPG08]	5 bits	Oui	Non
	Luo et coll. [LWBL09]	64 bits	Oui	Non
Techniques basées sur analyse multi-résolution	Kanai et coll. [KDK98]	$\approx$ 620 octets	Non	Non
	Uccheddu et coll. [UCB04]	1 bit	Oui	Non
	Praun et coll. [PHF99]	50 bits	Non	Oui
	Yin et coll. [YPSZ01]	250 bits	Non	Oui
Autres techniques	Bennour & Dugelay [BD06]	$\approx$ 500 bits	Non	Non
	Li et coll. [LZP*04]	24 bits	Semi	Non

TABLE A.4 – Continuation du Tableau A.3 : Résistance des différentes méthodes de tatouage robustes aux différents types d'attaques.

Algorithmes	Transform. similarité	Traitement signal	Deform. locale & coupe	Attaques connect.	Réarrange. éléments
Yu et coll. [YIK03]	Recalage	+	−	Rééchan.	Invariant
VFA [Ben99b]	+	−	−	−	Invariant
Zafeiriou et coll. [ZTP05]	+	+	−	+	Invariant
Cho et coll. [CPJ07]	+	+	−	+	Invariant
Bors [Bor06]	++	−	−	−−	Invariant
TSQ [OMA97]	++	−	+	−−	Invariant
Benedens [Ben99a]	Recalage	+	−	+	Invariant
Lee et coll. [LK07]	Recalage	+	−	+	Invariant
Ohbuchi et coll. [OMT02]	Recalage	++	++	Rééchan.	Invariant
Cayre et coll. [CRAS*03]	+	+	++	−−	Invariant
Wu & Kobbelt [WK05]	Recalage	++	++	Rééchan.	Rééchan.
Rondao-Alface & Macq [RAM05]	+	+	++	+	Invariant
Liu et coll. [LPG08]	++	+	−−	+	Invariant
Luo et coll. [LWBL09]	++	+	−−	+	Invariant
Kanai et coll. [KDK98]	+	−	−	−−	Invariant
Uccheddu et coll. [UCB04]	++	+	−	−	Invariant
Praun et coll. [PHF99]	Recalage	++	++	Rééchan.	Rééchan.
Yin et coll. [YPSZ01]	Recalage	+	−	Rééchan.	Rééchan.
Bennour & Dugelay [BD06]	Recalage	+	+	−	Invariant
Li et coll. [LZP*04]	+	+	+	Rééchan.	Invariant

Remarque : Dans ce tableau, «Rééchan.» signifie «Rééchantillonnage».

sont : les techniques spatiales sur les sommets, les techniques spatiales sur les facettes, les techniques basées sur l'analyse spectrale directe, les techniques basées sur l'analyse multi-résolution, et les autres techniques.

A.3.5 Discussion

Le tatouage de maillages 3D est un sujet de recherche intéressant et prometteur, avec de nombreuses applications potentielles. Cependant, en raison de nombreuses difficultés, la recherche dans ce domaine en est encore à son début. Dans cette section, nous avons présenté une étude bibliographique sur le tatouage de maillages. A partir de cette présentation, nous pouvons voir qu'il existe encore beaucoup de problèmes ouverts concernant ce sujet de recherche. Pour le tatouage fragile, un travail intéressant serait l'élaboration d'une méthode qui soit capable de localiser précisément les attaques subies et en même temps invariante à toutes les opérations «content-preserving». Pour le tatouage de haute capacité, il semble prometteur de combiner les principes des méthodes basées sur la géométrie et ceux des méthodes basées sur l'ordre des éléments combinatoires, pour construire de nouveaux algorithmes possédant les bonnes propriétés de ces deux types de méthodes. La communauté semble aussi très intéressée à trouver de nouvelles primitives spatiales de tatouage ainsi que des outils efficaces d'analyse spectrale, pour concevoir des méthodes robustes et aveugles plus performantes. Enfin, il est très important de construire un benchmark pour les techniques de tatouage de maillages 3D, afin de faciliter l'évaluation et la comparaison des différentes méthodes, et ainsi promouvoir la recherche dans ce domaine. Nous avons suivi ces pistes de recherche au cours du déroulement de cette thèse. Les résultats obtenus seront présentés dans les sections suivantes.

A.4 Schéma de Costa scalaire

Le schéma de Costa scalaire est une technique largement utilisée dans le tatouage aveugle des images, des fichiers audio et des vidéos. Dans cette section, nous présentons brièvement les processus d'insertion et d'extraction du tatouage adoptés par cette technique.

Sans perte de généralité, supposons que nous voulons cacher une séquence de symboles $w_i, i \in \{1, 2, ..., N\}$ dans une séquence de quantités scalaires $x_i, i \in \{1, 2, ..., N\}$. Chaque symbole w_i prend sa valeur dans l'alphabet $\mathcal{W} = \{0, 1, ..., R-1\}$ et peut ainsi transporter $\log_2 R$ bits. Dans la plupart des cas, nous choisissons $R = 2$, donc chaque symbole transporte exactement 1 bit ; par conséquent, le mécanisme de tatouage corres-

pondant est souvent appelé SCS binaire. Pour effectuer l'insertion du tatouage, d'abord nous construisons un dictionnaire structuré et pseudo-aléatoire $\mathcal{U}_{x_i,t_{x_i}}$ pour chaque quantité scalaire x_i comme montré ci-dessous :

$$\mathcal{U}_{x_i,t_{x_i}} = \bigcup_{l=0}^{R-1} \left\{ u = zS + l\frac{S}{R} + t_{x_i}S \right\}, \tag{A.5}$$

où $z \in \mathbb{Z}$ est un entier, S est le pas de quantification, $l \in \mathcal{W}$ est le symbole de la marque représenté par le mot de code u, et t_{x_i} est l'élément d'une séquence pseudo-aléatoire générée en utilisant une clé secrète K (cette séquence est aussi appelée le *signal dither*). Par exemple, t_{x_i} peut suivre une distribution uniforme entre $-\frac{1}{2}$ et $\frac{1}{2}$. Dans le contexte du tatouage, ce signal est introduit pour rendre les valeurs des mots de code u dans $\mathcal{U}_{d_i,t_{d_i}}$ pseudo-aléatoires, avec l'objectif de renforcer la sécurité de la méthode de tatouage. Notons que les mots de codes dans $\mathcal{U}_{d_i,t_{d_i}}$ représentent les symboles de tatouage dans l'alphabet $\mathcal{W} = \{0, 1, ..., R-1\}$, répartis façon uniforme et alternée.

Afin d'insérer un symbole de tatouage w_i en x_i, nous cherchons d'abord le mot de code u_{x_i} le plus proche de x_i dans le dictionnaire et qui représente correctement w_i. Cela signifie que w_i doit être égal à la valeur l dans la dérivation de u_{x_i} montrée dans l'équation (A.5). Ensuite, la valeur quantifiée $x_i^{'}$ est calculée comme ci-dessous :

$$x_i^{'} = x_i + \alpha \left(u_{x_i} - x_i \right), \tag{A.6}$$

où α est le *facteur de compensation de distorsion* qui prend généralement sa valeur entre 0 et 1, c.-à-d. $\alpha \in [0, 1]$. Le processus d'insertion consiste à pousser la valeur de x_i vers u_{x_i}, au moins dans l'intervalle $\left(u_{x_i} - \frac{S}{2R}, u_{x_i} + \frac{S}{2R}\right)$, qui est la zone de décodage de u_{x_i} sous le critère de la distance minimale.

Avec la connaissance de la clé secrète K et les valeurs des paramètres R (nombre de symboles dans l'alphabet $\mathcal{W}$) et S (pas de quantification) utilisées pendant l'insertion du tatouage, le message caché peut être extrait d'une manière aveugle. Premièrement, le dictionnaire pseudo-aléatoire $\mathcal{U}_{x_i,t_{x_i}}$ est construit pour chaque quantité scalaire reçue $\hat{x}_i$ à partir de laquelle nous voulons extraire un symbole de tatouage. Notons que la valeur de la primitive de tatouage $\hat{x}_i$ peut être différente de celle du signal tatoué initial $x_i^{'}$, en raison de l'existence des attaques, mais cela n'aura pas d'influence sur la construction du dictionnaire. En effet, les valeurs des mots de code dans le dictionnaire ne dépendent que des indices des primitives et sont indépendantes de ses valeurs réelles ; ainsi, nous pouvons assurer que, si les primitives de tatouage sont correctement synchronisées (c.-à-d. indexées), le même dictionnaire est obtenu lors de l'extraction. Ensuite, nous cher-

chons dans le dictionnaire établi $\mathcal{U}_{x_i,t_{x_i}}$ le mot de code $u_{\hat{x}_i}$ le plus proche de $\hat{x}_i$ sous le critère de la distance minimale :

$$u_{\hat{x}_i} = \arg \min_{u} \|\hat{x}_i - u\|, u \in \mathcal{U}_{x_i,t_{x_i}}. \tag{A.7}$$

Le symbole de tatouage extrait $\hat{w}_i$ est simplement le symbole représenté par le mot de code trouvé $u_{\hat{x}_i}$.

Les avantages principaux du schéma de Costa scalaire sont sa facilité d'implémentation, son efficacité en termes de temps de calcul et sa grande souplesse d'adaptation aux différents critères tels que la capacité, la distorsion, la robustesse et la sécurité. Dans les sections A.5-A.7, nous allons exploiter le SCS dans différents domaines de représentation pour construire des méthodes de tatouage aveugles. Notons que nous présenterons seulement les idées de base des méthodes proposées. Pour les détails des algorithmes ainsi que plus de résultats expérimentaux, le lecteur peut se référer aux chapitres correspondants (Chapitres 5-7, en anglais) de ce manuscrit de thèse.

A.5 Tatouage hiérarchique de maillages semi-réguliers basé sur la transformation en ondelettes

A.5.1 Motivation et système de tatouage proposé

Cette section présente un système de tatouage hiérarchique pour les maillages semi-réguliers. Trois tatouages différents (robuste, de haute capacité et fragile) sont insérés dans un même maillage semi-régulier, servant aux différentes applications (protection des copyright, enrichissement du contenu et authentification du contenu). En effet, les applications de tatouage ci-dessus ne sont pas mutuellement exclusives. Par exemple, nous pouvons imaginer le scénario suivant : un constructeur automobile élabore une pièce de véhicule complexe représentée par un maillage semi-régulier, puis il souhaite insérer dans cette pièce une information de copyright afin de protéger sa propriété intellectuelle ; il voudrait également insérer un tatouage fragile pour assurer que toute modification illégale sur cette pièce puisse être facilement détectée par les clients autorisés ; enfin, il pourrait s'intéresser à insérer dans la pièce des informations de description, telles que la norme de conception ou les modèles de voitures applicables, afin de faciliter l'utilisation de ce modèle.

A partir d'un maillage semi-régulier dense, l'analyse en ondelettes permet d'obtenir un maillage grossier qui représente la forme globale du maillage (basses fréquences) et une série de coefficients d'ondelettes qui représentent les informations de détails à

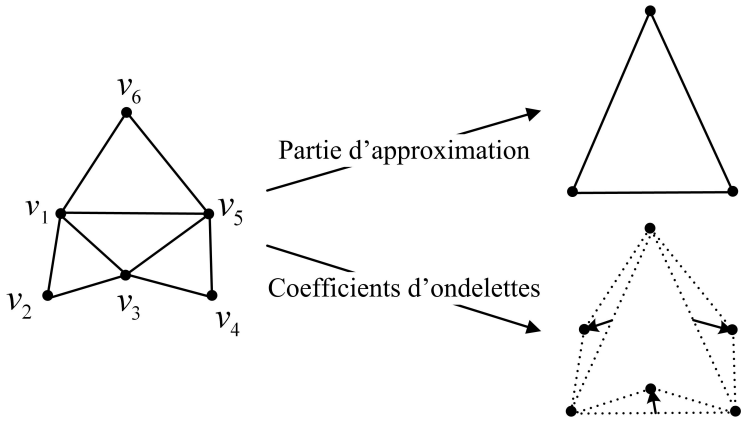


FIGURE A.5 – La décomposition en ondelettes paresseuses d'un maillage semi-régulier et triangulaire.

différents niveaux de résolution (moyennes et hautes fréquences). La formulation mathématique de l'analyse et de la synthèse en ondelettes d'un maillage 3D semi-régulier a été introduite par Lounsbery et coll. [LDW97]. La figure A.5 illustre une itération du mécanisme de décomposition en ondelettes paresseuses d'un maillage triangulaire semi-régulier. Un groupe de quatre triangles est fusionné en un seul triangle. Au cours de cette fusion, trois des six sommets (v_2, v_4, v_6 sur la figure A.5) sont conservés au niveau supérieur (maillage plus grossier), les trois autres (v_1, v_3, v_5) étant supprimés. Les coefficients d'ondelettes sont calculés comme les erreurs de prédiction pour les sommets supprimés : ce sont des vecteurs tridimensionnels associés à chaque arête du maillage plus grossier. La prédiction la plus simple est utilisée ici, à savoir le point central des deux sommets ayant été incidents au sommet supprimé. Une telle analyse peut être appliquée itérativement sur un maillage dense ayant une connectivité semi-régulière. L'algorithme de synthèse en ondelettes qui permet de reconstruire le maillage dense à partir de la représentation la plus grossière et l'ensemble des coefficients d'ondelettes est simple à définir.

L'analyse multi-résolution basée sur la transformation en ondelettes est un outil très approprié pour construire un système de tatouage multiple et hiérarchique : d'abord, il n'existe aucune interférence entre les différents tatouages s'ils sont insérés dans les vecteurs de coefficients d'ondelettes de niveaux différents ; ensuite, chaque tatouage peut être inséré dans le niveau de résolution le plus adapté à son fonctionnement. La figure A.6 illustre le système de tatouage hiérarchique proposé : le tatouage robuste est inséré en modifiant les normes des vecteurs des coefficients d'ondelettes associés au niveau de résolution le plus grossier ; le tatouage fragile est inséré dans le niveau de résolution dense obtenu après un niveau de décomposition en ondelettes du maillage origi-

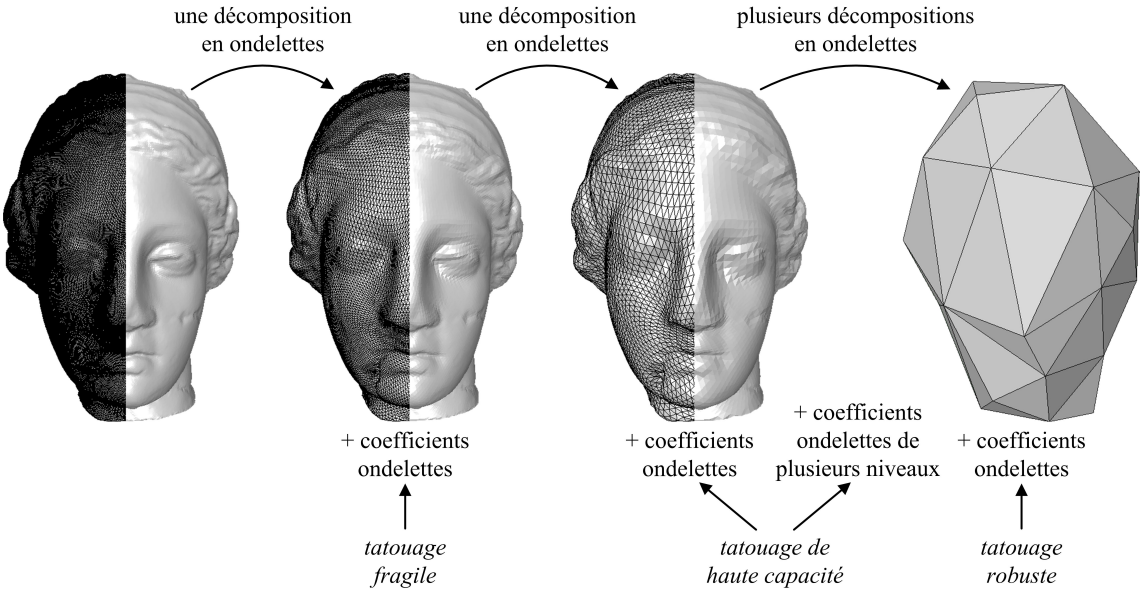


FIGURE A.6 – Le système de tatouage multiple et hiérarchique basé sur la transformation en ondelettes.

nal, par une modification des normes et des orientations des vecteurs des coefficients d'ondelettes ; le tatouage de haute capacité est inséré dans un ou plusieurs niveaux intermédiaires en considérant l'ordre des normes d'un groupe de vecteurs de coefficients d'ondelettes comme primitive de tatouage. En pratique, le tatouage robuste est inséré en premier après une décomposition complète, puis le tatouage de haute capacité et le tatouage fragile sont insérés successivement au cours du processus de reconstruction. Cet ordre de l'insertion évite au tatouage inséré postérieurement de dégrader le(les) tatouage(s) inséré(s) antérieurement. Les primitives de tatouage de ces trois méthodes sont toutes modifiées en utilisant la quantification de Costa scalaire présentée dans la section précédente.

A.5.2 Résultats et discussion

Le système de tatouage hiérarchique présenté ci-dessus a été implémenté et testé. Les résultats expérimentaux (cf. Chapitre 5) montrent l'efficacité des trois méthodes de tatouage intégrées dans le système. A notre connaissance, ce système constitue aussi la première tentative de tatouage multiple de maillages 3D. Dans les paragraphes suivants, nous présentons quelques discussions sur la conception et la performance des algorithmes de tatouage proposés.

Pendant l'élaboration du tatouage robuste, le problème de causalité et le problème de synchronisation sont soigneusement pris en compte. En comparaison avec les tatouages aveugles et robustes existants, l'avantage principal de notre algorithme est qu'il

peut introduire des modifications objectives relativement importantes tout en les gardant perceptuellement invisibles, car ces modifications sont plutôt de basse fréquence. Il est démontré que dans le cas du tatouage de maillages 3D, la modification de basse fréquence est à la fois plus imperceptible et plus robuste [SCOT03, ZvKD07]. Le tatouage robuste proposé possède expérimentalement une bonne résistance aux attaques géométriques telles que l'ajout de bruit, le lissage et la quantification des coordonnées des sommets. Par contre, la décomposition en ondelettes étant fortement liée à la connectivité du maillage, il s'avère, comme tous ses prédécesseurs dans cette catégorie, fragile aux attaques de connectivité.

En utilisant notre méthode de tatouage de haute capacité, il est facile d'insérer une grande quantité d'informations auxiliaires dans un maillage semi-régulier. Le tatouage inséré est invariant à la fois au réarrangement des éléments combinatoires et à la transformation de similarité. Cette méthode est un peu fragile aux attaques de géométrie de moyenne ou forte amplitude, mais en pratique dans la plupart des applications de tatouage de haute capacité, nous ne demandons pas la robustesse à ces attaques qui introduisent des modifications assez visibles. Techniquement, notre tatouage de haute capacité profite des avantages des méthodes basées sur la géométrie et des méthodes basées sur l'ordre des éléments combinatoires. En effet, le tatouage est inséré dans une primitive géométrique en appliquant l'idée de la stéganographie basée sur la permutation [Art01]. Comme montré dans le chapitre 5, la limite de capacité de notre méthode est approximativement égal à $3\left\lfloor \frac{N_0^v}{G} \right\rfloor . \lfloor \log_2 (G!) \rfloor$, où N_0^v est le nombre de sommets dans le maillage semi-régulier original et G est un entier qui contrôle le compromis entre la capacité et la robustesse. Cette capacité est beaucoup plus élevée que celles des méthodes basées sur la géométrie. L'avantage de cette technique par rapport aux méthodes basées sur l'ordre des éléments combinatoires est son invariance au réarrangement des sommets/facettes.

A notre connaissance, notre méthode fragile est la première dans la littérature qui soit robuste à toutes les opérations «content-preserving» et en même temps capable de localiser avec une grande précision les attaques considérées comme non-tolérables à l'égard de l'intégrité du maillage tatoué. Nous avons choisi comme primitives de tatouage deux quantités géométriques locales des vecteurs de coefficient d'ondelettes, toutes deux invariantes à la transformation de similarité. Ces deux primitives peuvent être quantifiées d'une manière indépendante pour y insérer des symboles de la marque ; donc le problème de causalité est évité. L'intégrité de la primitive d'authentification (arête dans le maillage semi-régulier obtenu après une décomposition en ondelettes) repose sur l'égalité des symboles insérés dans les deux primitives de tatouage qui lui sont associées.

La principale limitation de notre système de tatouage multiple et hiérarchique est qu'il peut être appliqué seulement sur des maillages avec une connectivité semi-régulière, car la transformation en ondelettes régulières ne peut être effectuée que sur ce type particulier de maillages. A l'avenir, nous voudrions étudier la possibilité de construire un tel système de tatouage multiple pour des maillages de connectivité arbitraire.

A.6 Tatouage robuste et aveugle de maillages 3D basé sur les moments volumiques

Dans cette section, nous nous intéressons au tatouage robuste et aveugle de maillages 3D de connectivité quelconque. Dans la méthode proposée, un tatouage aveugle multi-bits est inséré dans le maillage hôte en modifiant légèrement ses moments volumiques locaux via une quantification de type SCS «adaptative».

A.6.1 Moments volumiques

Le moment volumique d'une surface 3D fermée est défini comme l'intégration volumique suivante :

$$m_{pqr} = \int\int\int x^p y^q z^r \rho\,(x,y,z)\,dxdydz, \tag{A.8}$$

où p, q, r sont les ordres, et $\rho\,(x,y,z)$ est la fonction d'indication de volume (elle est égale à 1 si (x,y,z) est à l'intérieur de la surface ; sinon elle est égale à 0). Pour un maillage orientable, Zhang et Chen [ZC01] et Tuzikov et coll. [TSV03] ont dérivé l'expression explicite pour cette intégration volumique. L'idée est de la calculer comme une somme d'intégrations primaires sur des primitives géométriques élémentaires. Pour un maillage triangulaire, la primitive est le tétraèdre construit sur les sommets d'une facette triangulaire f_i et l'origine du système de coordonnées $\mathcal{O}$. Le signe de la contribution pour chaque intégration primaire est déterminé selon l'orientation de f_i et la position relative entre f_i et $\mathcal{O}$. Notons que si les facettes sont correctement orientées, le moment m_{000} est le volume de la surface fermée. Avec ce calcul, les moments volumiques peuvent être facilement généralisés pour les surfaces orientables mais non-fermées (par exemple des «patchs» de maillage 3D). Le calcul consiste à ajouter des facettes imaginaires en reliant les sommets de bord et l'origine $\mathcal{O}$, puis à calculer les moments pour la surface fermée obtenue. Ces moments volumiques ont des caractéristiques robustes et sont souvent utilisés pour le recalage et l'indexation de maillages [ZC01].

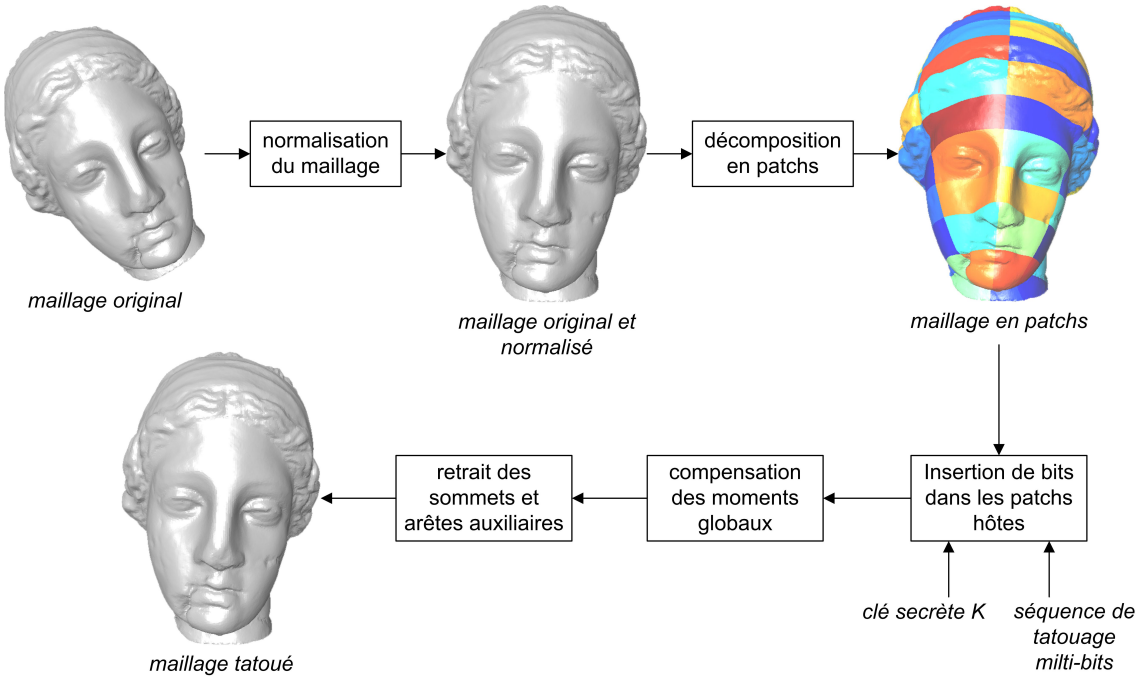


FIGURE A.7 – Le processus d'insertion du tatouage basé sur les moments volumiques.

A.6.2 Méthode de tatouage proposée

La méthode robuste et aveugle proposée dans cette section est basée sur l'hypothèse qu'un tatouage robuste de maillages 3D doit être lié à la *forme 3D* intrinsèque du maillage, mais pas à ses éléments combinatoires (c.-à-d. ses sommets, arêtes et facettes). De ce point de vue, les moments analytiques et continus présentés ci-dessus semblent de bons candidats pour être les primitives de tatouage. Ces descripteurs sont de nature continue et ne dépendent que de la forme 3D analytique représentée par le maillage. Ainsi, ils devraient être robustes aux attaques de géométrie, de connectivité et même aux conversion de représentations d'objets 3D (par exemple de maillage à voxels) à condition que ces attaques ne modifient pas trop la forme du maillage tatoué. Nous avons souhaité utiliser les moments volumiques pour insérer un tatouage lisible et multi-bits. Il surgit immédiatement deux difficultés : premièrement, les moments de différents ordres sont corrélés, donc il est difficile de modifier les différents moments d'un maillage simultanément et indépendamment pour insérer plusieurs bits ; deuxièmement, la transformation pour obtenir ces moments n'est pas réversible, donc nous sommes obligés de les modifier indirectement dans le domaine spatial en déplaçant les sommets. Le premier point nous a forcés à décomposer le maillage en plusieurs patchs et à insérer un bit dans chaque patch. Concernant le deuxième point, nous introduisons un algorithme itératif pour la déformation des patchs.

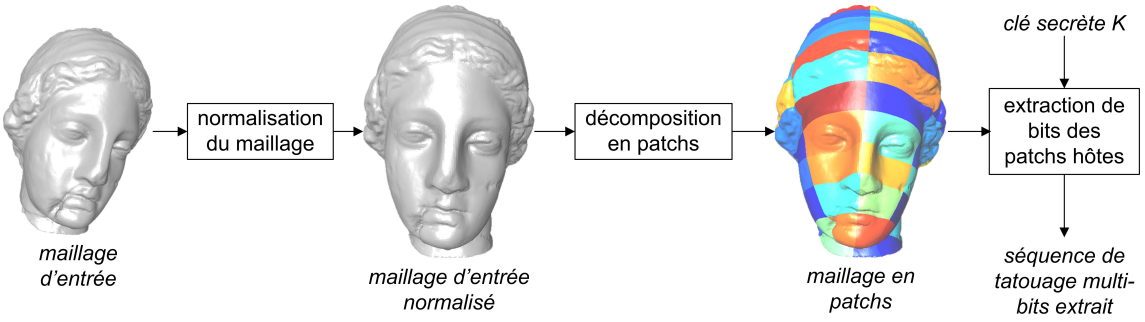


FIGURE A.8 – Le processus d'extraction du tatouage basé sur les moments volumiques.

La figure A.7 illustre le processus d'insertion du tatouage. Le maillage hôte est d'abord normalisé en utilisant ses moments volumiques globaux. Puis, il est transformé du système de coordonnées Cartésien (x, y, z) au système cylindrique (h, r, θ). Le maillage est ensuite décomposé en patchs par une simple discrétisation de ses domaines h et θ. Pour certains patchs hôtes sélectionnés, nous calculons les moments d'ordre zéro et les quantifions pour insérer un bit par patch. Afin d'assurer un calcul précis pour les moments des patchs, nous insérons des sommets et arêtes auxiliaires sur les bords des patchs ; ils peuvent être facilement retirés après l'insertion du tatouage. La modification des moments est effectuée par une déformation itérative des patchs, avec l'utilisation d'un masque lisse pour assurer l'imperceptibilité du tatouage et l'absence de transitions brutales aux niveau des raccords entre les patchs. Une troisième difficulté, le problème de causalité, apparaît à ce stade. En effet, après la déformation des patchs hôtes, les moments globaux du maillage sont généralement modifiés ; par conséquent, nous risquons de ne pas trouver, d'une manière aveugle, la même pose du maillage à l'extraction après la normalisation. Un processus de compensation des moments globaux est introduit pour résoudre ce problème.

La Figure A.8 illustre le processus d'extraction du tatouage aveugle. Ce processus consiste successivement en la normalisation du maillage, sa décomposition en patchs et l'extraction du bit inséré dans chacun des patchs hôtes obtenus.

A.6.3 Quelques résultats expérimentaux

Nous avons testé notre algorithme sur différents maillages 3D. Les résultats expérimentaux et les comparaisons avec les méthodes de l'état de l'art montrent la supériorité de notre méthode en termes d'imperceptibilité et de robustesse (surtout aux attaques de connectivité, même spatialement anisotropes). En outre, à notre connaissance, notre travail est aussi la première tentative dans la littérature pour lutter contre les attaques de changement de représentations des objets 3D (la discrétisation du maillage en

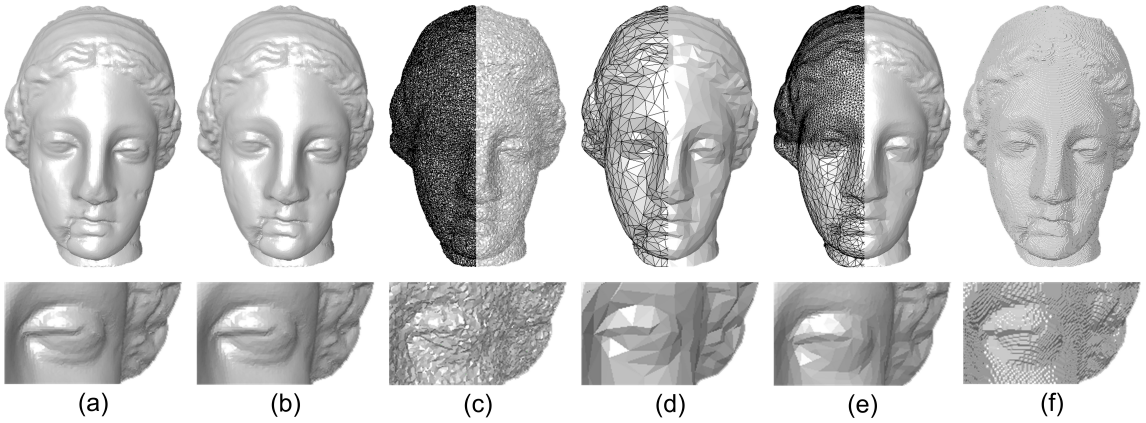


FIGURE A.9 – Résultats expérimentaux du tatouage robuste et aveugle basé sur les moments volumiques : (a) Venus original ; (b) Venus tatoué (75 bits sont insérés), il n'existe presque pas de distorsion visuelle ; (c) Venus tatoué et attaqué par un bruit additif aléatoire de 0.50%, 89% de bits sont correctement extraits ; (d) attaqué par une forte simplification uniforme (97.5% de sommets sont enlevés), 93% de bits sont correctement extraits ; (e) attaqué par une forte simplification non-uniforme (75% de sommets sont enlevés), 91% de bits sont correctement extraits ; (f) attaqué par un changement de représentations (discrétisation en $350 \times 350 \times 350$ voxels, et puis triangulation Marching Cubes), 87% de bits sont correctement extraits.

voxels). La figure A.9 illustre quelques résultats expérimentaux pour le maillage Venus. La bonne performance de notre méthode repose sur la stabilité intrinsèque des moments volumiques globaux et locaux du maillage. En effet, les moments globaux sont utilisés pour réaliser une normalisation robuste du maillage, et les moments locaux sont utilisés comme primitives de tatouage.

A.7 Tatouage robuste et aveugle de maillages 3D basé sur la transformation en harmoniques variétés

A.7.1 Objectif

En général, les méthodes spectrales de tatouage de maillages 3D ont l'avantage d'être plus imperceptibles. En effet, le système visuel humain est moins sensible aux modifications des composants de basse et moyenne fréquences d'un maillage 3D. Par conséquent, un tatouage inséré dans ces composants est moins visible pour l'œil humain. En outre, après une transformation inverse du domaine spectral au domaine spatial, les modifications introduites dans le domaine spectral par l'insertion du tatouage sont propagées à toutes les parties du maillage. Ainsi, il est moins possible pour une méthode de tatouage spectral d'introduire des distorsions localisées visibles sur la surface du maillage (par exemple avec des formes spécifiques). Dans le sens général, cela aide également à

renforcer la sécurité du tatouage car il devient plus difficile pour un pirate de remarquer l'existence d'un tatouage ou de localiser les bits insérés.

Notre objectif dans cette section est d'utiliser la technique de quantification pour insérer un tatouage aveugle et robuste dans le domaine spectral en harmoniques variétés [VL07, VL08]. Nous voudrions améliorer la capacité de tatouage dans ce domaine prometteur (actuellement de 5 bits dans la méthode de Liu et coll. [LPG08]), tout en préservant le mieux possible les autres critères de performance (c.-à-d. la robustesse, l'imperceptible, la sécurité et l'efficacité en termes de temps de calcul). Dans la méthode proposée, un tatouage aveugle de 16 bits est inséré en utilisant une quantification SCS itérative des coefficients spectraux de basse fréquence obtenus par la transformation en harmoniques variétés.

A.7.2 Transformation en harmoniques variétés

Pour effectuer la transformation en harmoniques variétés d'un maillage triangulaire, nous avons à résoudre l'équation matricielle suivante :

$$-Q\mathbf{h}^k = \lambda_k D\mathbf{h}^k, \tag{A.9}$$

où $\mathbf{h}^k = \left[H_1^k, H_2^k, ..., H_n^k\right]^T$; la matrice diagonale D, appelée matrice de masse concentrée, est de dimension $n \times n$ (n étant le nombre de sommets du maillage) avec $D_{i,i} = \left(\sum_{t \in \mathcal{N}_t(i)} |t|\right)/3$; Q, appelée matrice de dureté, est aussi de dimension $n \times n$ avec

$$\begin{cases} Q_{i,j} = \left(\cot(\beta_{i,j}) + \cot(\beta'_{i,j})\right)/2, \\ Q_{i,i} = -\sum_j Q_{i,j}. \end{cases} \tag{A.10}$$

Dans les expressions ci-dessus, $\mathcal{N}_t(i)$ représente l'ensemble des triangles incidents au sommet v_i, l'opérateur $|.|$ est la surface d'un triangle, et $\beta_{i,j}, \beta'_{i,j}$ sont les deux angles opposés à l'arête reliant v_i et v_j. Les vecteurs propres $\mathbf{h}^k$ de l'équation (A.9) sont les bases en harmoniques variétés, tandis que les valeurs propres représentent leurs fréquences associées. Les bases sont orthogonales sous le produit interne fonctionnel. Après calcul, les bases spectrales sont normalisées pour qu'elles aient toutes des normes unités et ordonnées selon l'ordre croissant de leurs fréquences associées. Les coefficients spectraux sont calculés comme le produit interne fonctionnel entre la géométrie $\mathbf{x}$ (resp. $\mathbf{y}, \mathbf{z}$) du maillage et les bases orthonormales et mises en ordre :

$$\tilde{x}_k = \left\langle \mathbf{x}, \mathbf{h}^k \right\rangle = \sum_{i=1}^{n} x_i D_{i,i} H_i^k. \tag{A.11}$$

Finalement, la k-ième amplitude spectrale est définie comme :

$$c_k = \sqrt{(\tilde{x}_k)^2 + (\tilde{y}_k)^2 + (\tilde{z}_k)^2}. \tag{A.12}$$

Le maillage peut être exactement reconstruit en utilisant la transformation inverse en harmoniques variétés. Pour la géométrie $\mathbf{x}$ (resp. $\mathbf{y}, \mathbf{z}$), nous avons

$$x_i = \sum_{k=1}^{n} \tilde{x}_k H_i^k. \tag{A.13}$$

Les premiers 100 ou 200 coefficients spectraux de basse fréquence peuvent être calculés en utilisant un algorithme spécifique «band-by-band» [VL07, VL08] combiné avec un solveur efficace comme TAUCS [TCR03] or SuperLU [DGL09]. Par exemple, les premiers 100 coefficients du maillage Lapin possédant 33.5K sommets peuvent être obtenus en moins de 40 secondes sur un ordinateur portable ordinaire. Contrairement aux coefficients spectraux obtenus par une analyse spectrale Laplacienne combinatoire [KG00], les coefficients spectraux en harmoniques variétés sont très robustes aux attaques de connectivité. Cet avantage rend le domaine spectral en harmoniques variétés très prometteur pour le tatouage robuste de maillages.

A.7.3 Algorithme de tatouage

Le processus d'insertion du tatouage se compose de trois étapes :

1. D'abord, nous transformons le maillage hôte de l'espace spatial à l'espace spectral en utilisant l'équation (A.11) avec les bases en harmoniques variétés du maillage ;
2. Ensuite, nous quantifions une partie des amplitudes spectrales de basse fréquence en utilisant le schéma de Costa scalaire binaire (cf. Section A.4), pour insérer avec redondance, un tatouage de 16-bit ;
3. Enfin, nous reconstruisons le maillage tatoué avec les coefficients spectraux modifiés en utilisant l'équation (A.13).

La quantification des amplitudes spectrales est très simple, mais il existe deux problèmes critiques importants. Le premier est le problème de causalité. Nous pouvons constater que les bases en harmoniques variétés du maillage reconstruit ne sont pas les mêmes que celles du maillage original, car la géométrie du maillage a été modifiée lors de l'insertion de tatouage. Cela signifie que si nous faisons une nouvelle transformation en harmoniques variétés sur le maillage reconstruit, nous obtiendrons des coefficients spectraux différents de ceux souhaités (c.–à-d. ceux obtenus à l'insertion après tatouage).

D'une manière similaire à la méthode de Liu et coll. [LPG08], nous adoptons la solution la plus simple pour résoudre ce problème, à savoir la quantification itérative des amplitudes spectrales. Plus précisément, nous prenons le maillage reconstruit comme maillage à tatouer et re-quantifions les amplitudes spectrales plusieurs fois jusqu'à ce que tous les bits puissent être correctement extraits. Pour réduire le nombre d'itérations, nous avons trois règles. Premièrement, nous ne quantifions pas les 20 premières amplitudes car leurs modifications ont expérimentalement une influence assez importante sur les amplitudes aux fréquences successives. Deuxièmement, à partir de c_{21}, seuls 3 coefficients sur 4 sont quantifiés. Ainsi, nous créons des «espaces d'amortissement» entre les coefficients à quantifier, ce qui alléger efficacement le problème de causalité. La dernière mesure est d'essayer d'insérer le tatouage de 16 bits en le répétant 3 fois. Avec cette répétition, nous pouvons arrêter le processus itératif même s'il existe encore des erreurs de quantification sur quelques amplitudes «difficiles», à condition que ces erreurs ne soient pas trop nombreuses et que les votes de tous les bits répétés soient corrects. Les coefficients c_{21} à c_{84} sont doc utilisés pour insérer 16 bits. Grâce à ces trois mesures, notre méthode est expérimentalement applicable sur les «gros» maillages ayant plus de 10K sommets, avec des temps d'exécution acceptables, c'est-à-dire inférieurs à 10 minutes.

Le deuxième problème est l'invariance à la transformation de similarité, qui inclut la translation, la rotation et la mise à l'échelle uniforme. Sous une translation, nous pouvons démontrer que seulement c_1 est modifiée. Comme c_1 ne participe pas au processus d'insertion, les bits insérés restent constants après translation. Il est aussi prouvé qu'une rotation dans l'espace spatial provoque la même rotation dans l'espace spectral, ce qui n'a aucune influence sur les amplitudes spectrales c_k, d'où l'invariance à la rotation. Sous une mise à l'échelle uniforme avec facteur s, nous pouvons démontrer que les c_k sont aussi mis à l'échelle avec un facteur s^2. Pratiquement, nous fixons le pas de quantification pour les $c_k, 21 \leq k \leq 84$ comme βc_2, avec β une constante. Ainsi, les mots de code dans le dictionnaire de quantification pour c_k changent proportionnellement avec c_k sous une mise à l'échelle uniforme, et donc nous gardons bien l'invariance du tatouage.

A.7.4 Quelques résultats expérimentaux

Notre méthode spectrale de tatouage a été testée sur plusieurs objets et comparée avec deux méthodes récentes. Globalement, son point fort réside dans l'imperceptibilité du tatouage. La figure A.10 illustre quelques exemples de maillages originaux et tatoués, ainsi que les cartes de distorsion objective entre eux. Nous ne constatons presque aucune

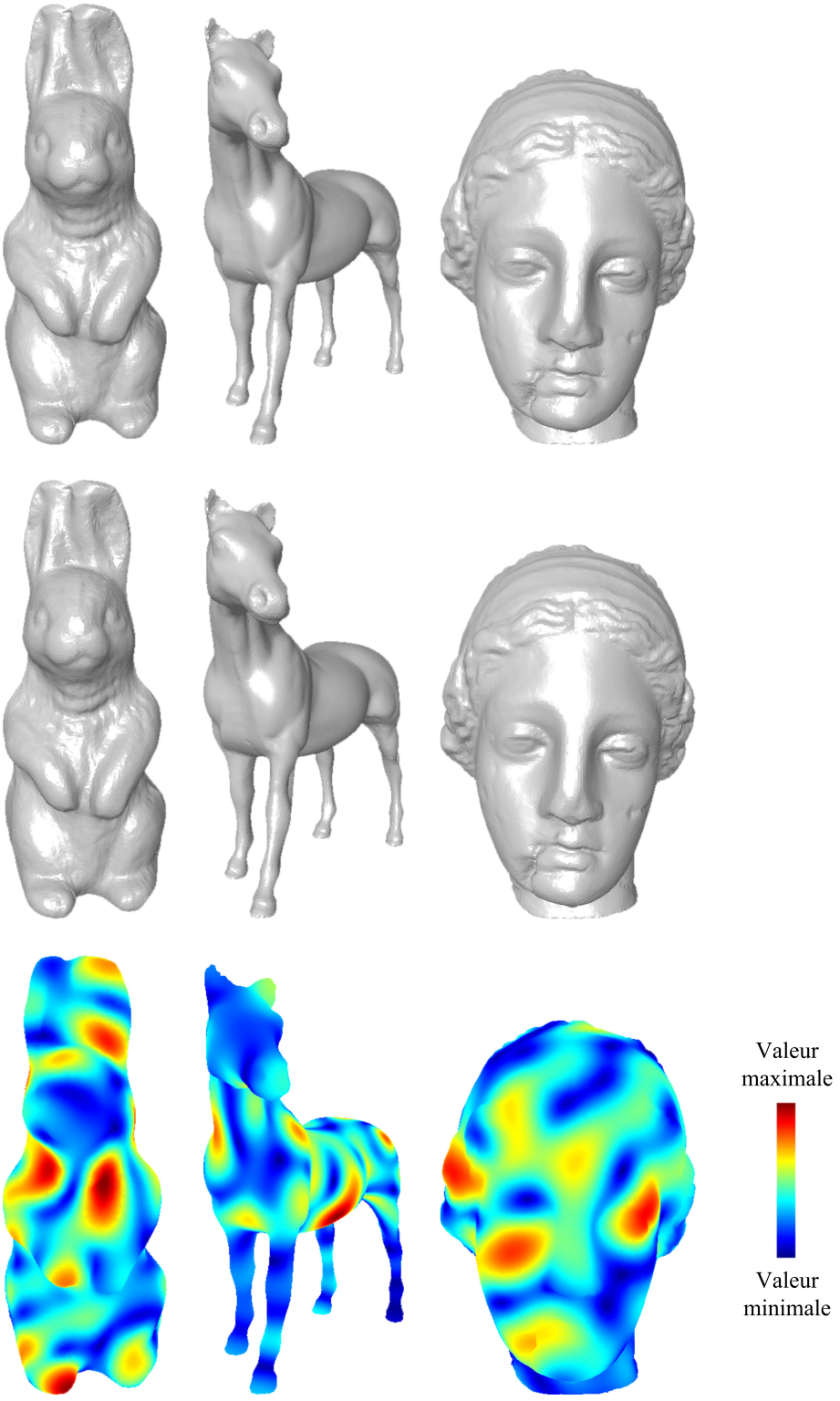


FIGURE A.10 – Dans la première colonne sont les maillages originaux ; dans la deuxième colonne sont les maillages tatoués avec 16 bits insérés ; les cartes de distorsion objective correspondantes sont illustrées dans la dernière colonne.

différence visuelle entre un modèle original et sa version tatouée, parce que la modification introduite par l'insertion du tatouage est de basse fréquence. Les comparaisons avec les deux méthodes récentes sont effectuées sous les pré-conditions d'une robustesse globale comparable et d'une même capacité. Par rapport à la méthode spatiale de Cho et coll. [CPJ07], notre méthode spectrale nécessite des temps d'insertion et d'extraction plus longs, mais ce désavantage est compensé par une meilleure imperceptibilité et aussi une meilleure robustesse aux attaques de connectivité. En comparant avec la méthode de Liu et coll. [LPG08] qui est aussi basée sur la transformation en harmoniques variétés, notre méthode possède un meilleur compromis entre la distorsion et la robustesse ainsi qu'une capacité plus élevée (jusqu'à 16 bits contre 5 bits pour la méthode de Liu et coll.).

Nous avons également constaté quelques défauts de la méthode proposée. En effet, cette méthode peut échouer sur certains maillages : soit les coefficients spectraux du maillage ne sont pas assez robustes, soit nous n'arrivons pas à insérer le tatouage même après beaucoup d'itérations. Nous voudrions trouver des explications théoriques pour ces échecs et si possible, une solution pour régler ce problème. Nous nous intéressons également à augmenter la capacité de notre méthode, à renforcer sa robustesse, et à trouver une solution efficace et élégante pour le problème de causalité.

A.8 Un benchmark pour les techniques de tatouage robustes de maillages 3D

A.8.1 Motivation

Lorsqu'une nouvelle méthode de tatouage robuste de maillages 3D est proposée par la communauté scientifique, les auteurs souhaitent habituellement la comparer avec les méthodes existantes pour évaluer équitablement ses points forts et ses points faibles. Cependant, il semble actuellement difficile d'effectuer une comparaison efficace entre différentes techniques, principalement en raison du fait que les auteurs des différentes méthodes utilisent souvent des modèles 3D différents, des mesures de distorsion différentes, des attaques différentes et enfin des méthodologies d'évaluation différentes lors de leurs études expérimentales. La conséquence négative de cette situation est que nous sommes obligés de ré-implémenter et/ou de re-tester les méthodes existantes si nous voulons effectuer une comparaison expérimentale. La ré-implémentation et les tests nécessitent beaucoup de temps, et constituent donc une difficulté supplémentaire pour les chercheurs de ce domaine. De plus, cette ré-implémentation des méthodes existantes

risque de générer des résultats contestables puisqu'elle est forcément légèrement différente de l'implémentation originale de ses auteurs. Notre objectif dans cette section est de construire un système de benchmark pour les techniques de tatouage robustes de maillages 3D, afin de faciliter les comparaisons expérimentales entre les différentes méthodes existantes et à venir.

Notre système de benchmark évalue une méthode selon trois aspects : la capacité, la distorsion et la robustesse. Nous ne tenons pas compte de la sécurité, car la recherche en tatouage de maillages en est encore à ses débuts (cf. la section A.3) et jusqu'à présent la communauté s'intéresse plutôt à l'élaboration de méthodes aveugles capables de résister aux attaques de connectivité plutôt qu'à la sécurité de l'algorithme qui est plutôt considérée comme une demande de haut niveau. Enfin, lors de la présentation des résultats d'évaluation, les auteurs doivent également indiquer si leur méthode est aveugle, semi-aveugle ou non-aveugle.

A.8.2 Le système de benchmark

Le système de benchmark proposé comprend une collection «standard» de maillages 3D, un outil logiciel et deux protocoles d'évaluation. L'outil logiciel permet de calculer les distorsions objectives et perceptuelles introduites par l'insertion du tatouage, et également d'exercer diverses attaques sur un maillage tatoué. Les deux protocoles d'évaluation sont destinés à deux types d'applications différents. Ces protocoles indiquent les étapes principales à suivre lors de l'évaluation expérimentale d'une technique de tatouage.

Collection «standard» de maillages 3D

Nous avons sélectionné plusieurs maillages représentatifs (comprenant des nombres de sommets et des complexités différentes). Ces modèles sont : Bunny (34835 sommets), Venus (100759 sommets), Cheval (112642 sommets), Dragon (50000 sommets) et Lapin (70658 sommets). Nous avons également acquis les permissions (conférées par Stanford Computer Graphics Laboratory et Cyberware Inc.) pour les rendre téléchargeables librement sur notre serveur public à `http://liris.cnrs.fr/meshbenchmark/`.

Outil logiciel

Nous choisissons l'erreur quadratique maximum ou «maximum root mean square error» notée MRMS [CRS98, ASCE02] pour mesurer la distorsion objective introduite par l'insertion de tatouage. La MRMS est une distance continue de surface à surface, et assure donc une haute précision de calcul de distorsion. La mesure structurelle de

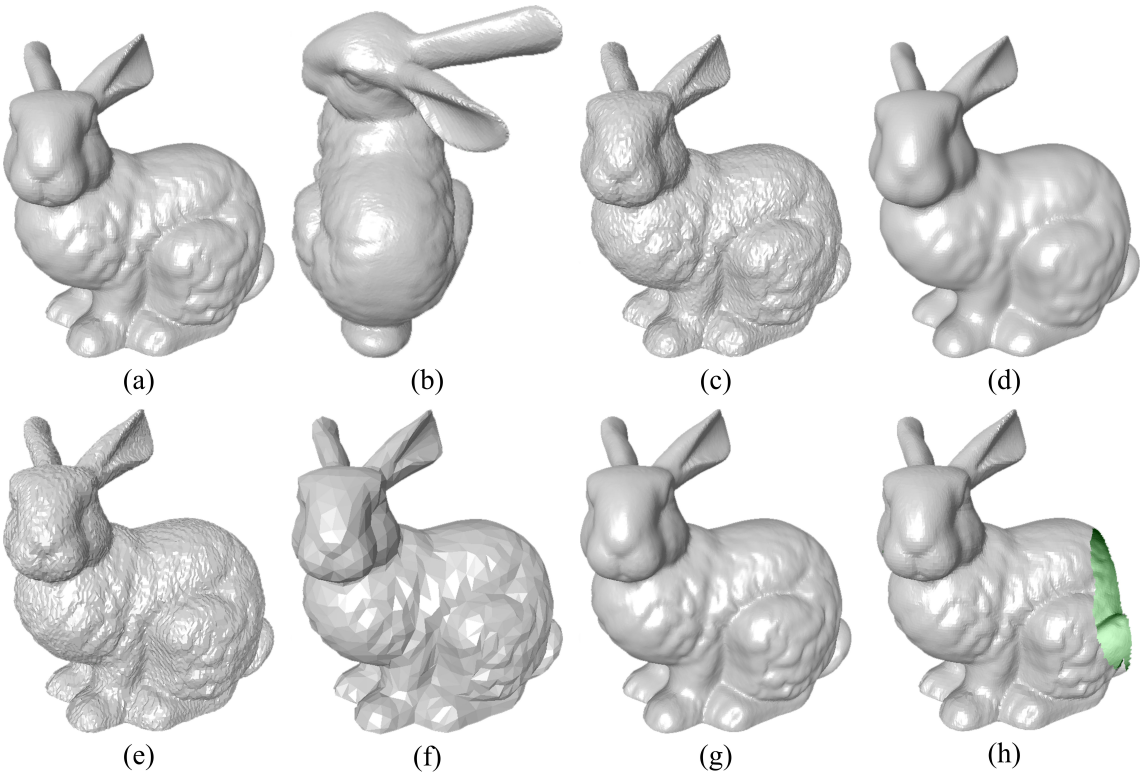


FIGURE A.11 – Le modèle Stanford Bunny et quelques versions attaquées obtenues en utilisant notre outil logiciel compris dans le système de benchmark : (a) le maillage original possédant 34835 sommets et 104499 arêtes ; (b) après une transformation de similarité ; (c) après un ajout de bruit ($A = 0.30\%$) ; (d) après un lissage Laplacian ($\lambda = 0.10, N_{itr} = 30$) ; (e) après une quantification des coordonnées des sommets ($R = 8$) ; (f) après une simplification ($E_{sim} = 95\%$) ; (g) après une subdivision (1 itération, schéma de Loop) ; (h) après une coupe ($V_{cr} = 10\%$).

distorsion ou «mesh structural distortion measure» (MSDM) est utilisée pour mesurer la distorsion perceptuelle introduite. La principale raison du choix de la mesure MSDM est sa haute cohérence avec les résultats subjectifs donnés par les êtres humains [LGD*06]. Notre logiciel intègre les implémentations des calculs de ces deux distorsions fournies par P. Cignoni et G. Lavoué.

Le logiciel inclut également les implémentations d'une grande quantité d'attaques courantes. La figure A.11 illustre quelques modèles Bunny attaqués, qui ont été générés par notre logiciel.

- **Attaque de fichier** : le réarrangement des listes de sommets et de facettes dans le fichier de données.
- **Transformation de similarité** : translation, rotation, mise à l'échelle uniforme et leur combinaison.
- **Ajout de bruit** : le paramètre réglable est l'amplitude du bruit A.

- **Lissage Laplacien** : le facteur de déformation λ est fixé 0.10, le nombre d'itérations N_{itr} varie.
- **Quantification des coordonnées des sommets** : le paramètre réglable est le niveau de quantification R (en bits).
- **Simplification** : nous avons choisi l'algorithme de Lindstrom et Turk [LT98] ; le paramètre réglable est le ratio de réduction d'arêtes E_{sim}.
- **Subdivision** : une itération de subdivision est appliquée en utilisant différents mécanismes incluant le schéma «midpoint», le schéma $\sqrt{3}$ et le schéma de Loop [ZSoo].
- **Coupe** : le paramètre réglable est le ratio approximatif de réduction de sommets V_{cr}.

Protocoles d'évaluation

Nous proposons deux protocoles d'évaluation qui définissent les principales étapes à suivre lors de l'étude expérimentale des techniques de tatouage robustes. Le premier est le protocole orienté «qualité perceptuelle» et le second est le protocole orienté «qualité géométrique». Notre motivation pour créer deux protocoles différents est que les différentes applications utilisant les maillages 3D n'ont pas les mêmes restrictions sur les distorsions objectives et perceptuelles introduites par l'insertion du tatouage. Par exemple, pour les maillages utilisés dans les loisirs numériques, nous devons d'abord assurer que la distorsion introduite n'est pas visuellement gênante (c.-à-d. que le modèle tatoué doit avoir une qualité visuelle très élevée), tandis que l'amplitude de la distorsion objective est moins importante. Au contraire, pour les maillages utilisés dans la conception assistée par ordinateur et l'imagerie médicale, il est souvent nécessaire d'assurer une distorsion objective très faible, alors que la qualité visuelle du modèle tatoué est relativement moins importante.

Le protocole d'évaluation orienté «qualité perceptuelle» comprend les étapes suivantes :

1. Insérer un tatouage W dans un maillage de test $\mathcal{M}$ en utilisant une clé secrète K pour obtenir un modèle tatoué $\mathcal{M}'$; s'assurer que la distorsion perceptuelle $d_{MSDM} \leq 0.20$ et que la distorsion objective $d_{MRMS} \leq 0.08\%.l_{bbd}$, où l_{bbd} représente la longueur de la diagonale de la boîte englobante du maillage.
2. Appliquer les attaques listées dans le tableau A.5 sur le maillage tatoué $\mathcal{M}'$, en utilisant l'outil logiciel compris dans le système de benchmark.

TABLE A.5 – Attaques utilisées dans les protocoles d'évaluation.

Attaque	Paramètre	Valeurs de paramètre
Attaque de fichier	fois	3
Transformation de similarité	fois	3
Ajout de bruit*	A	0.05%, 0.10%, 0.30%, 0.50%
Lissage ($\lambda = 0.10$)	N_{itr}	5, 10, 30, 50
Quantization	R	11, 10, 9, 8, 7
Simplification	E_{sim}	10%, 30%, 50%, 70%, 90%, 95%, 97.5%**
Subdivision (1 itération)	schéma	midpoint, $\sqrt{3}$, Loop
Coupe	V_{cr}	10%, 30%, 50%

* Pour chaque amplitude, il est nécessaire de répéter l'ajout de bruit pour 3 fois.
** Le ratio 97.5% est seulement pour les maillages avec plus de 100K sommets.

3. Essayer d'extraire (où de détecter) le tatouage inséré W des maillages tatoués attaqués et enregistrer les résultats de robustesse de l'algorithme d'extraction (où de détection) du tatouage.
4. Répéter les étapes 1-3 5 fois avec différentes séquences de tatouage et des clés secrètes aléatoires.
5. Répéter les étapes 1-4 pour chaque modèle de la collection «standard» de maillages 3D fourni dans le système de benchmark.

Le protocole orienté «qualité géométrique» comprend les mêmes étapes ; la différence réside au niveau des contraintes sur les distorsions objective et perceptuelle : $d_{MRMS} \leq 0.02\%.l_{bbd}$ et $d_{MSDM} \leq 0.30$. Nos protocoles sont capables de tester les méthodes lisibles mais également les méthodes détectables. Pour les méthodes lisibles, nous suggérons de répéter l'insertion du tatouage au moins 5 fois sur chaque modèle et de rapporter les moyennes des ratios de bits erronés des tatouages extraits sous les différentes attaques comme résultats de l'évaluation. Pour les méthodes détectables, il est recommandé que, pour chaque modèle de test, l'insertion du tatouage soit répétée au moins 100 fois en utilisant différentes séquences de tatouage et différentes clés secrètes ; les courbes de caractéristiques de fonctionnement du récepteur relatives à chaque type d'attaque sont ensuite rapportées comme résultats de l'évaluation. Enfin, pour les méthodes lisibles, nous proposons de fixer la capacité du tatouage comme l'une des valeurs suivantes : 16 bits, 32 bits, 64 bits et $\geq$ 96 bits.

Nos méthodes robustes basées sur les ondelettes (cf. section A.5), les moments volumiques (cf. section A.6) et l'analyse spectrale (cf. section A.7), ainsi que la méthode spatiale de Cho et coll. [CPJ07] ont été évaluées et comparées dans le cadre du système de benchmark proposé. Le lecteur peut se référer au chapitre 8 pour les résultats quantitatifs obtenus. La conclusion est que par rapport à la méthode de Cho et coll., nos trois

méthodes montrent une meilleure performance dans les applications qui demandent une très bonne qualité perceptuelle du modèle tatoué. De plus, parmi toutes les techniques comparées, notre méthode basée sur les moments volumiques possède en général la meilleure robustesse aux attaques de connectivité, pour les deux types d'applications.

A.9 Conclusion

A.9.1 Résumé des contributions

Dans cette thèse, nous avons présenté nos travaux de recherche sur le tatouage numérique de maillages 3D. Notre objectif principal était de construire plusieurs méthodes aveugles efficaces. En effet, les trois types de tatouages aveugles de maillages 3D (robuste, fragile et de haute capacité) ont tous des applications prometteuses (protection de copyright, authentification du contenu et enrichissement du contenu). Notre objectif principal a été accompli par la dérivation de plusieurs algorithmes de tatouage aveugles dans différents domaines de représentation, tous basés sur la quantification scalaire de Costa. Les résultats expérimentaux ont démontré l'efficacité des méthodes proposées en termes de robustesse, imperceptibilité et capacité. Notre objectif secondaire était de fournir un outil de benchmark pour les techniques de tatouage robustes de maillages, afin de faciliter les évaluations et les comparaisons des différentes méthodes. Cet objectif secondaire a été atteint par la construction et la mise en ligne d'un système de benchmark open-source.

Dans cette thèse, nous avons apporté les contributions suivantes :

Etude bibliographique exhaustive avec un point de vue centré sur les attaques
Nous avons présenté les méthodes de tatouage de maillages existantes en les classifiant comme fragiles, haute capacité et robustes. Pour chaque type de techniques, nous avons clairement défini son objectif et ses contraintes. Après cette présentation classique de l'état de l'art, une investigation centrée sur les attaques a été menée (cf. Chapitre 3). Les attaques possibles sur les maillages tatoués ont été classifiées, et les contre-mesures existantes pour résister à chaque type d'attaque ont été analysées et discutées. La motivation de cette investigation est que les attaques jouent un rôle très important lors de l'élaboration d'une méthode de tatouage pour les maillages. En effet, l'une des difficultés spécifiques pour le tatouage de maillages, par rapport au tatouage des images, audio et vidéo, est l'existence de nombreuses attaques particulières auxquelles il est, en même temps, difficile de résister. Nous espérons que ce nouveau point de vue sur l'état de l'art sera utile pour mieux comprendre les difficultés rencontrées et pour découvrir des

directions de recherche prometteuses.

Introduction du schéma de Costa scalaire pour le tatouage de maillages 3D

Le schéma de Costa scalaire (SCS) est une technique de quantification largement utilisée dans le tatouage aveugle des images, sons et vidéos. Les principaux avantages de cette technique sont son implémentation facile et sa grande souplesse entre la capacité, la distorsion, la robustesse et la sécurité. Dans cette thèse, nous avons introduit le schéma de Costa scalaire pour le tatouage des maillages 3D. Dans la méthode basée sur les moments volumiques, nous avons également légèrement modifié le SCS original afin de le rendre plus adaptable à l'espace d'insertion de tatouage spécifique (cf. Chapitre 6). En utilisant ce schéma, nous avons élaboré plusieurs algorithmes de tatouage aveugles dans trois domaines de représentation différents : le domaine d'ondelettes pour des maillages semi-réguliers, les domaines spatial et spectral pour les maillages de connectivité quelconque. Les primitives de tatouage, qui sont soumises à la quantification scalaire, sont respectivement les normes et les orientations des vecteurs de coefficients d'ondelettes, les moments volumiques locaux et les amplitudes spectrales en harmoniques variétés.

Tatouage multiple de maillages semi-réguliers

Il est parfois nécessaire d'insérer plusieurs tatouages différents dans un même contenu multimédia pour différents services ou applications. Pour les maillages semi-réguliers, nous avons proposé un système de tatouage multiple et hiérarchique basé sur la transformation en ondelettes. Trois tatouages différents (robuste, de haute capacité et fragile) ont été insérés à différents niveaux de résolution d'un même maillage semi-régulier. A notre connaissance, ce travail constitue la première tentative de ce type dans la littérature.

Tatouage robuste et aveugle basé sur un descripteur de forme 3D continu

Le moment volumique analytique et continu est un descripteur efficace de forme 3D. Il est prouvé que les valeurs de ces moments restent très stables sous différentes attaques à condition que ces attaques ne modifient pas trop la forme intrinsèque du maillage. Nous avons proposé une technique de tatouage robuste et aveugle dans le domaine spatial en utilisant les moments volumiques locaux comme primitives de tatouage. Les points forts de cette méthode sont sa très bonne imperceptibilité et sa forte robustesse contre les attaques de connectivité. A notre connaissance, cette méthode est le premier algorithme de tatouage de maillages 3D qui utilise un descripteur de forme de nature continue comme primitive de tatouage. De plus, ce travail constitue également la première tentative dans la littérature pour atteindre la robustesse aux attaques de conversion de représentations

d'objets 3D (discrétisation du maillage en voxels).

Tatouage robuste et aveugle dans un domaine spectral
Nous avons également proposé une méthode de tatouage robuste et aveugle dans le domaine spectral en utilisant la transformation en harmoniques variétés. Les amplitudes spectrales obtenues à l'aide de cette transformation sont très robustes aux diverses attaques, y compris aux changements de connectivité. Un tatouage multi-bits est inséré dans ce nouveau domaine prometteur en quantifiant itérativement les amplitudes spectrales de basse fréquence du maillage hôte. La méthode proposée est efficace en termes de temps de calcul et en même temps robuste aux attaques de connectivité. Il existe encore peu de techniques spectrales qui possèdent ces deux propriétés.

Un système de benchmark accessible en ligne
Enfin, nous avons mis en place un système de benchmark pour l'évaluation des techniques de tatouage robustes de maillages 3D. Ce benchmark comprend une collection «standard» de modèles 3D, un outil logiciel et de deux protocoles d'évaluation. Le système de benchmark, qui est un projet open-source, a été rendu accessible librement sur Internet à `http://liris.cnrs.fr/meshbenchmark/`. A notre connaissance, c'est le premier benchmark de tatouage de maillages proposé à la communauté scientifique.

Une caractéristique importante des travaux présentés dans cette thèse est que nous avons construit des méthodes de tatouage aveugles efficaces en combinant des outils issus de différents domaines de recherche. Ces outils comprennent le schéma de Costa scalaire qui provient de la recherche en tatouage numérique, le moment volumique plutôt utilisé en analyse de forme 3D, et la transformation en harmoniques variétés qui provient de la recherche en traitement géométrique. Cette méthodologie de recherche interdisciplinaire pourrait être aussi considérée comme une contribution. En effet, nous sommes convaincus que le futur de la recherche en tatouage de maillages sera fortement lié aux avancements en traitement géométrique et analyse de forme 3D (cf. la section suivante qui présente les perspectives), et que beaucoup de travaux intéressants seront réalisés si les experts de ces différents domaines peuvent collaborer plus activement entre eux.

A.9.2 Perspectives

En ce qui concerne les perspectives, nous distinguons les travaux futurs à court terme et les directions de recherche à long terme. Les travaux à court terme concernent les améliorations des travaux présentés dans les différents chapitres de ce manuscrit.

Améliorations du système de tatouage multiple de maillages semi-réguliers
Pour le tatouage robuste, nous comptons utiliser dans notre méthode un code correcteur d'erreurs avancé (au lieu de la simple répétition de bits) pour la séquence de tatouage, afin de renforcer la robustesse et/ou d'augmenter la capacité. Pour le tatouage de haute capacité, nous voudrions concevoir un code correcteur d'erreurs pour le codage basé sur la permutation, afin de rendre la méthode moins fragile. Nous souhaitons également tester l'idée d'utiliser les propriétés géométriques locales du maillage pour synchroniser les tatouages robuste et de haute capacité.

Améliorations de la méthode basée sur les moments
Tout d'abord, il est nécessaire de chercher la raison pour la forte déformation de certains patchs au cours de l'insertion du tatouage et ensuite d'effectuer quelques rectifications de notre algorithme pour équilibrer les distorsions dans les différentes zones du maillage. De plus, nous nous intéressons à développer un mécanisme de décomposition de maillage adaptatif et robuste qui produit des patchs de tailles comparables. Ce mécanisme de décomposition «intelligent» pourrait permettre d'insérer plus de bits dans le maillage sans dégrader l'imperceptibilité et la robustesse du tatouage ; il pourrait également aider à résoudre le problème de désynchronisation entraîné par la classification des patchs (cf. Chapitre 6 pour plus de détails).

Améliorations de la méthode spectrale
Premièrement, il serait intéressant de trouver une explication théorique pour la robustesse des coefficients spectraux calculés par la transformation en harmoniques variétés. Comme mentionné dans la section A.7, notre méthode peut échouer pour certains objets en raison du problème de robustesse des coefficients spectraux ou du problème de causalité. Ainsi, il semble nécessaire d'effectuer une étude expérimentale pour mieux comprendre le comportement des coefficients spectraux sous diverses attaques et sous la modulation des amplitudes spectrales due à l'insertion du tatouage. Avec les résultats de cette étude, nous pourrons effectuer plusieurs améliorations de notre méthode. Par exemple, une meilleure robustesse peut être atteinte si nous évitons d'insérer les bits dans les coefficients qui sont intrinsèquement moins stables. Enfin, une meilleure performance globale peut être acquise si nous trouvons une solution efficace et élégante pour le problème de causalité.

Améliorations du système de benchmark
Nous envisageons de continuer notre travail sur le benchmark en fournissant davantage de maillages de test et davantage d'attaques. En effet, il serait intéressant d'intégrer les

attaques basées sur l'estimation [VPP*01, PVM*01] dans le benchmark et de tester la résistance des méthodes existantes contre ces attaques plus «intelligentes» et aussi plus destructrices. En même temps, nous espérons recevoir des retours de la communauté de recherche, sur lesquels nous pourrons nous baser pour améliorer notre système de benchmark et ainsi renforcer son utilité.

De façon plus générale, il existe encore beaucoup de problèmes ouverts dans le domaine du tatouage de maillages. Les études sur ces problèmes constitueront nos travaux futurs à moyen et long terme.

Utiliser d'autres descripteurs de forme 3D pour le tatouage robuste et aveugle

Il semble évident que les descripteurs de forme 3D pourraient être des primitives de tatouage très efficaces. En effet, les histogrammes utilisés dans les méthodes de Zafeiriou et coll. [ZTP05] et de Cho et coll. [CPJ07] sont des descripteurs de forme statistiques. Le moment volumique utilisé dans notre méthode aveugle et robuste décrite dans la section A.6 est un descripteur de forme basé sur une transformation. Par conséquent, il semble intéressant d'explorer la possibilité d'utiliser d'autres descripteurs de forme 3D pour le tatouage de maillages robuste et aveugle. Quelques exemples de ces descripteurs prometteurs sont les moments de Zernike 3D [NK04], les coefficients de transformation radiale angulaire 3D [RCB05] et les coefficients de transformation en harmoniques sphériques [FMK*03]. Certains d'entre eux sont particulièrement intéressants en raison de leur invariance intrinsèque à la rotation et leur robustesse aux diverses opérations. Mais malheureusement, les descripteurs ci-dessus sont tous définis sur les objets 3D discrétisés en voxels. Par conséquent, il semble que nous devrons d'abord dériver ces descripteurs pour les maillages 3D et ensuite construire des algorithmes de tatouage efficaces en les utilisant comme primitives de tatouage.

La robustesse à la coupe combinée avec des changements de connectivité

Cette opération est considérée comme l'attaque la plus sévère pour un tatouage robuste et aveugle. Il semble qu'il existe deux solutions possibles : la première est d'introduire un pré-traitement de segmentation «robuste et aveugle» du maillage qui soit capable de résister à cette attaque, puis d'insérer le tatouage répétitivement dans chaque patch du maillage segmenté ; la seconde solution est d'utiliser, comme primitive de tatouage, un descripteur de forme local robuste, dont la valeur reste stable sous cette attaque. Lors de la recherche sur ces deux solutions, la communauté scientifique pourrait bénéficier des avancées récentes dans la recherche en analyse et indexation de formes 3D, tels que les travaux de Shapira et coll. [SSCO08] et de Liu et coll. [LZSCO09].

Tatouage de maillages adaptatif

La performance d'une méthode de tatouage peut être améliorée si elle tient compte des propriétés locales du maillage hôte. Par exemple, dans les régions avec une densité d'échantillonnage moins élevé ou avec une haute rugosité, nous pouvons renforcer l'intensité de l'insertion pour un tatouage robuste ou augmenter le nombre de bits insérés pour un tatouage de haute capacité.

Tatouage de haute capacité invariant à toutes les opérations «content-preserving»

Dans de nombreuses applications, il est nécessaire qu'un tatouage de haute capacité soit invariant à la fois au réarrangement des éléments combinatoires et à la transformation de similarité, tout en gardant un grand nombre de bits insérés. Afin d'atteindre cet objectif, dans notre méthode de haute capacité basée sur les ondelettes (cf. Section A.5), nous insérons le tatouage dans des primitives géométriques en appliquant l'idée de la stéganographie basée sur la permutation. Il serait intéressant d'étudier s'il est possible d'élaborer une méthode similaire pour les maillages arbitraires.

Tatouage fragile efficace pour les maillages arbitraires

Dans la section A.5 nous avons proposé une méthode fragile efficace pour les maillages semi-réguliers, mais il semble beaucoup plus difficile de construire une telle méthode pour les maillages arbitraires. La difficulté principale est d'obtenir en même temps les propriétés suivantes : immunité au problème de causalité, invariance à toutes les opérations «content-preserving», niveau de sécurité élevée et stabilité numérique.

Tatouage de maillages invariant à la déformation naturelle forte

Un maillage 3D peut être soumis à certaines déformations fortes et réalistes pour former une séquence de maillages dynamiques. Par exemple, nous pouvons faire courir le modèle de Cheval, ou faire prendre au modèle Venus différentes expressions faciales. Idéalement, un tatouage robuste doit être capable de résister à ce type de déformations. Une solution possible est de trouver un descripteur de forme 3D invariant à ces déformations fortes pour l'utiliser comme primitive de tatouage. Une nouvelle fois, nous pourrions bénéficier des avancements dans la recherche en analyse de forme 3D, en particulier les travaux sur la représentation de formes invariante aux déformations naturelles [EK03, JZ07, Rus07].

Quelques exemples d'autres questions ouvertes sont : Quel est le «meilleur» domaine spectral pour le tatouage robuste et aveugle de maillages ? Comment et où pouvons-nous ajouter une fonctionnalité de tatouage dans une chaîne de compression de maillages ?

Est-il possible de protéger par tatouage les maillages progressifs ? Est-il possible d'utiliser le tatouage pour la compression de maillages ou de séquences de maillages ? Trouver des réponses à ces questions pourrait aussi constituer des directions de recherche intéressantes.

Pour conclure, nous pensons que le tatouage de maillages 3D est un sujet de recherche très intéressant, avec beaucoup d'applications potentielles et de nombreux problèmes ouverts. Nous sommes convaincus que ce domaine de recherche a un avenir très prometteur, surtout avec l'effort conjoint des experts des différents domaines tels que le tatouage numérique, l'analyse de forme 3D et le traitement géométrique, la théorie de l'information et de la communication.

Bibliography

[ABB07] Abdallah E. E., Ben Hamza A., Bhattacharya P.: Spectral graph-theoretic approach to 3D mesh watermarking. In *Proceedings of the Graphics Interface Conference* (2007), pp. 327–334. 36, 45

[ABB08] Abdallah E. E., Ben Hamza A., Bhattacharya P.: Robust 3D watermarking technique using eigendecomposition and nonnegative matrix factorization. In *Proceedings of the International Conference on Image Analysis and Recognition* (2008), pp. 253–262. 36, 44, 45, 47

[AE03] Ashourian M., Enteshary R.: A new masking method for spatial domain watermarking of three-dimensional triangle meshes. In *Proceedings of the IEEE Conference on Convergent Technologies for Asia-Pacific Region* (2003), vol. 1, pp. 428–431. 44

[AEJ04] Ashourian M., Enteshari R., Jeon J.: Digital watermarking of three-dimensional polygonal models in the spherical coordinate system. In *Proceedings of the Computer Graphics International* (2004), pp. 590–593. 31

[AF06] Attene M., Falcidieno B.: ReMESH: An interactive environment to edit and repair triangle meshes. In *Proceedings of the IEEE International Conference on Shape Modeling and Applications* (2006), pp. 271–276. 19, 117, 181

[AGL06] Ahn M. S., Guskov I., Lee S. Y.: Out-of-core remeshing of large polygonal meshes. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006), 1221–1228. 67

[Art01] Artz D.: Digital steganography: Hiding data within data. *IEEE Internet Computing 5*, 3 (2001), 75–80. 26, 76, 191

[ASCE02] Aspert N., Santa-Cruz D., Ebrahimi T.: MESH: Measuring error between surfaces using the Hausdorff distance. In *Proceedings of the IEEE International Conference on Multimedia & Expo* (2002), pp. 705–708. 151, 201

[ATo4] Alghoniemy M., Tewfik A. H.: Geometric invariance in image watermarking. *IEEE Transactions on Image Processing 13*, 2 (2004), 145–153. 99

[AUGA08] Alliez P., Ucelli G., Gotsman C., Attene M.: *Shape Analysis and Structuring*. Springer-Verlag, 2008, ch. Recent advances in remeshing of surfaces, pp. 53–82. 38

[Aur91] Aurenhammer F.: Voronoi diagrams — A survey of a fundamental geometric data structure. *ACM Computer Survey 23*, 3 (1991), 345–405. 56

[Bar97] Barton J. M.: Method and apparatus for embedding authentication information within digital data, 1997. *United States Patent 5646997*. 52

[BBoo] Benedens O., Busch C.: Towards blind detection of robust watermarks in polygonal models. *Computer Graphics Forum 19*, 3 (2000), C199–C208. 43

[BBo4] Barni M., Bartolini F.: *Watermarking Systems Engineering: Enabling Digital Assets Security and other Applications*. Marcel Dekker Inc., 2004. 7, 11, 12, 178

[BDo6] Bennour J., Dugelay J. L.: Protection of 3D object visual representations. In *Proceedings of the IEEE International Conference on Multimedia & Expo* (2006), pp. 1113–1116. 33, 49, 185

[BDo7] Bennour J., Dugelay J. L.: Toward a 3D watermarking benchmark. In *Proceedings of the IEEE International Workshop on Multimedia Signal Processing* (2007), pp. 369–372. 149

[Ben99a] Benedens O.: Geometry-based watermarking of 3D models. *IEEE Computer Graphics and Applications 19*, 1 (1999), 46–55. 31, 32, 46, 49, 185

[Ben99b] Benedens O.: Two high capacity methods for embedding public watermarks into 3D polygonal models. In *Proceedings of the ACM Multimedia and Security Workshop* (1999), pp. 95–99. 29, 31, 49, 76, 185

[BGIo8] Bogomjakov A., Gotsman C., Isenburg M.: Distortion-free steganography for polygonal meshes. *Computer Graphics Forum 27*, 2 (2008), 637–642. 26, 28, 184

[BGML96] Bender W., Gruhl D., Morimoto N., Lu A.: Techniques for data hiding. *IBM Systems Journal 35*, 3–4 (1996), 313–336. 52

[Big93] Biggs N.: *Algebraic Graph Theory (Second Edition)*. Cambridge University Press, 1993. 34

[Bor06] Bors A. G.: Watermarking mesh-based representations of 3-D objects using local moments. *IEEE Transactions on Image Processing 15*, 3 (2006), 687–701. 32, 41, 49, 98, 185

[BPK*07] Botsch M., Pauly M., Kobbelt L., Alliez P., Lévy B., Bischoff S., Rössl C.: Geometric modeling based on polygonal meshes. In *Proceedings of the ACM Siggraph Course Notes* (2007). 7, 11

[Cay07] Cayre F.: Watermarking fundamentals, 2007. *Course Notes for Graduate Students of INP Grenoble.* Available at `http://www.balistic-lab.org/pub/ens/wm/wm-fundamentals-M2SCCI.pdf` (access date: 14 July 2009). xiii, 57

[CCR08] Cignoni P., Corsini M., Ranzuglia G.: MeshLab: An open-source 3D mesh processing system. *ERCIM News*, 73 (2008), 45–46. 19, 181

[CFF05] Cayre F., Fontaine C., Furon T.: Watermarking security: Theory and practice. *IEEE Transactions on Signal Processing 53*, 10 (2005), 3976–3987. 62

[CGEB07] Corsini M., Gelasca E. D., Ebrahimi T., Barni M.: Watermarked 3-D mesh quality assessment. *IEEE Transactions on Multimedia 9*, 2 (2007), 247–255. 84, 152

[CKLS97] Cox I. J., Kilian J., Leighton T., Shamoon T.: Secure spread spectrum watermarking for multimedia. *IEEE Transactions on Image Processing 6*, 12 (1997), 1673–1687. 52

[CLLP05] Cho W. H., Lee M. E., Lim H., Park S. Y.: Watermarking technique for authentication of 3-D polygonal meshes. In *Proceedings of the International Workshop on Digital Watermarking* (2005), pp. 259–270. 24, 25, 43, 183

[CM03] Cayre F., Macq B.: Data hiding on 3-D triangle meshes. *IEEE Transactions on Signal Processing 51*, 4 (2003), 939–949. xiii, 26, 27, 28, 41, 76, 184

[CMB*07] Cox I. J., Miller M. L., Bloom J. A., Fridrich J., Kalker T.: *Digital Watermarking and Steganography (Second Edition)*. Morgan Kaufmann Publishers Inc., 2007. 7, 11, 87, 178

[CMM99] Cox I. J., Miller M. L., Mckellips A. L.: Watermarking as communications with side information. *Proceedings of the IEEE 87*, 7 (1999), 1127–1141. 54

[Cos83] Costa M.: Writing on dirty paper. *IEEE Transactions on Information Theory 29*, 3 (1983), 439–441. xiii, 20, 54, 55, 181

[CPFPG05] Comesaña P., Pérez-Freire L., Pérez-González F.: Fundamentals of data hiding security and their application to spread-spectrum analysis. In *Proceedings of the International Workshop on Information Hiding* (2005), pp. 146–160. 62

[CPGR00] Chou J., Pradhan S., Ghaoui L. E., Ramchandran K.: A robust optimization solution to the data hiding problem using distributed source coding principles. In *Proceedings of the SPIE Electronic Imaging* (2000), vol. 3974, pp. 270–279. 55

[CPJ07] Cho J. W., Prost R., Jung H. Y.: An oblivious watermarking for 3-D polygonal meshes using distribution of vertex norms. *IEEE Transactions on Signal Processing 55*, 1 (2007), 142–155. xiii, xiv, 30, 31, 44, 46, 49, 97, 102, 121, 132, 140, 148, 150, 159, 169, 185, 200, 204, 209

[CRAS*03] Cayre F., Rondao-Alface P., Schmitt F., Macq B., Maître H.: Application of spectral decomposition to compression and watermarking of 3D triangle mesh geometry. *Signal Processing: Image Communication 18*, 4 (2003), 309–319. xiii, 36, 37, 45, 46, 49, 132, 185

[CRS98] Cignoni P., Rocchini C., Scorpigno R.: Metro: Measuring error on simplified surfaces. *Computer Graphics Forum 17*, 2 (1998), 167–174. 84, 151, 201

[CT06] Chou C. M., Tseng D. C.: A public fragile watermarking scheme for 3D model authentication. *Computer-Aided Design 38*, 11 (2006), 1154–1165. 21, 22, 25, 82, 183

[CW98] Chen B., Wornell G. W.: Digital watermarking and information embedding using dither modulation. In *Proceedings of the IEEE International Workshop on Multimedia Signal Processing* (1998), pp. 273–278. 55

[CW01a] Chen B., Wornell G. W.: Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Transactions on Information Theory 47*, 4 (2001), 1423–1443. xiii, 54, 56, 57, 108

[CW01b] Chen B., Wornell G. W.: Quantization index modulation methods for digital watermarking and information embedding of multimedia. *Journal of VLSI Signal Processing Systems 27*, 1-2 (2001), 7–33. 56

[CW06] Cheng Y. M., Wang C. M.: A high-capacity steganographic approach for 3D polygonal meshes. *The Visual Computer 22*, 9 (2006), 845–855. 26, 28, 41, 77, 184

[CW07] Cheng Y. M., Wang C. M.: An adaptive steganographic algorithm for 3D polygonal meshes. *The Visual Computer 23*, 9 (2007), 721–732. 26, 28, 184

[CWPG04] Cotting D., Weyrich T., Pauly M., Gross M.: Robust watermarking of point-sampled geometry. In *Proceedings of the Shape Modeling International* (2004), pp. 233–242. 35

[DF88] Donnelly H., Fefferman C.: Nodal sets of eigenfunctions on Riemannian manifolds. *Inventiones Mathematicae 93*, 1 (1988), 161–183. 34

[DFS05] Dodgson N. A., Floater M. S., Sabin M. A.: *Advances in Multiresolution for Geometric Modelling*. Springer-Verlag, 2005. 23

[DGL09] Demmel J. W., Gilbert J. R., Li X. S.: SuperLU users' guide, 2009. Software available at `http://crd.lbl.gov/~xiaoye/SuperLU/` (access date: 14 July 2009). 136, 197

[Dix84] Dixon R.: *Spread Spectrum Systems*. John Wiley & Sons, 1984. 52

[EBTG03] Eggers J. J., Bauml R., Tzschoppe R., Girod B.: Scalar costa scheme for information embedding. *IEEE Transactions on Signal Processing 51*, 4 (2003), 1003–1019. 4, 51, 56, 61, 62, 107, 138, 176

[EK03] Elad A., Kimmel R.: On bending invariant signatures for surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence 25*, 10 (2003), 1285–1295. 170, 210

[ESG00] Eggers J. J., Su J. K., Girod B.: A blind watermarking scheme based on structured codebooks. In *Proceedings of the IEE International Seminar on Secure Images and Image Authentication* (2000), pp. 4/1–4/21. 55

[FH05] Floater M. S., Hormann K.: *Advances in Multiresolution for Geometric Modelling*. Springer-Verlag, 2005, ch. Surface parameterization: A tutorial and survey, pp. 157–186. 39, 105

[Fli97] Flikkema P. G.: Spread-spectrum techniques for wireless communications. *IEEE Signal Processing Magazine 14*, 3 (1997), 26–36. 52

[FMK*03] Funkhouser T., Min P., Kazhdan M., Chen J., Halderman A., Dobkin D., Jacobs D.: A search engine for 3D models. *ACM Transactions on Graphics 22*, 1 (2003), 83–105. 169, 209

[GGS03] Gotsman C., Gu X., Sheffer A.: Fundamentals of spherical parameterization for 3D meshes. In *Proceedings of the ACM Siggraph* (2003), pp. 358–363. 40

[GGvL96] Golub G. H., Gene H., van Loan C. F.: *Matrix Computations (Third Edition)*. Johns Hopkins University Press, 1996. 36

[GH97] Garland M., Heckbert P. S.: Surface simplification using quadric error metrics. In *Proceedings of the ACM Siggraph* (1997), pp. 209–216. 117

[GJ93] Gray R. M., Jr. T. G. S.: Dithered quantizers. *IEEE Transactions on Information Theory 39*, 3 (1993), 805–812. 58, 60

[GN98] Gray R. M., Neuhoff D. L.: Quantization. *IEEE Transactions on Information Theory 44*, 6 (1998), 2325–2383. 55

[GSS99] Guskov I., Sweldens W., Schröder P.: Multiresolution signal processing for meshes. In *Proceedings of the ACM Siggraph* (1999), pp. 325–334. 39

[Gus07] Guskov I.: Manifold-based approach to semi-regular remeshing. *Graphical Models 69*, 1 (2007), 1–18. 67

[GVSS00] Guskov I., Vidimce K., Sweldens W., Schröder P.: Normal meshes. In *Proceedings of the ACM Siggraph* (2000), pp. 95–102. 83

[HB02] Harte T., Bors A. G.: Watermarking 3D models. In *Proceedings of the IEEE International Conference on Image Processing* (2002), vol. 3, pp. 661–664. 32, 41

[HG98] Hartung F., Girod B.: Watermarking of uncompressed and compressed video. *Signal Processing 66*, 3 (1998), 283–301. 52

[Hop96] Hoppe H.: Progressive mesh. In *Proceedings of the ACM Siggraph* (1996), pp. 99–108. 39, 40

[Hor84] Horn B. K. P.: Extended Gaussian images. *Proceedings of the IEEE 72*, 2 (1984), 1671–1686. 31

[JDBP04] Jin J. Q., Dai M. Y., Bao H. J., Peng Q. S.: Watermarking on 3D mesh based on spherical wavelet transform. *Journal of Zhejiang University: Science 5*, 3 (2004), 251–258. 38

[JZ07] Jain V., Zhang H.: A spectral approach to shape-based retrieval of articulated 3D models. *Computer-Aided Design 39*, 5 (2007), 398–407. 170, 210

[KDK98] Kanai S., Date H., Kishinami T.: Digital watermarking for 3D polygons using multiresolution wavelet decomposition. In *Proceedings of the International Workshop on Geometric Modeling: Fundamentals and Applications* (1998), pp. 296–307. 38, 43, 49, 185

[KG00] Karni Z., Gotsman C.: Spectral compression of mesh geometry. In *Proceedings the ACM Siggraph* (2000), pp. 279–286. 34, 131, 197

[KKL*03] Kwon K. R., Kwon S. G., Lee S. H., Kim T. S., Lee K. I.: Watermarking for 3D polygonal meshes using normal vector distributions of each patch. In *Proceedings of the IEEE International Conference on Image Processing* (2003), vol. 2, pp. 499–502. 32, 46

[KL03] Kim H. S., Lee H. K.: Invariant image watermark using Zernike moments. *IEEE Transactions on Circuits and Systems for Video Technology 13*, 8 (2003), 766–775. 99

[KP99] Kutter M., Petitcolas F. A. P.: A fair benchmark for image watermarking systems. In *Proceedings of the SPIE Electronic Imaging* (1999), vol. 3657, pp. 226–239. 28, 104, 184

[KP00] Katzenbeisser S., Petitcolas F. A.: *Information Hiding Techniques for Steganography and Digital Watermarking*. Artech House Inc., 2000. 7, 11, 178

[KSS00] Khodakovsky A., Schröder P., Sweldens W.: Progressive geometry compression. In *Proceedings of the ACM Siggraph* (2000), pp. 271–278. 67

[KTP03] Kalivas A., Tefas A., Pitas I.: Watermarking of 3D models using principal component analysis. In *Proceedings of the IEEE International Conference on Acoustic, Speech, and Signal Processing* (2003), vol. 1, pp. 637–640. 43, 102

[KVJP05] Kim M. S., Valette S., Jung H. Y., Prost R.: Watermarking of 3D irregular meshes based on wavelet multiresolution analysis. In *Proceedings of the International Workshop on Digital Watermarking* (2005), pp. 313–324. 39

[Lav09] Lavoué G.: A local roughness measure for 3D meshes and its application to visual masking. *ACM Transactions on Applied Perception 5*, 4 (2009), 1–23. 73, 83

[LB08] Luo M., Bors A. G.: Principal component analysis of spectral coefficients for mesh watermarking. In *Proceedings of the IEEE International Conference on Image Processing* (2008), pp. 441–444. 36, 131

[LC87] Lorensen W. E., Cline H. E.: Marching cubes: A high resolution 3D surface construction algorithm. In *Proceedings of the ACM Siggraph* (1987), pp. 163–170. 120

[LDD07] Lavoué G., Denis F., Dupont F.: Subdivision surface watermarking. *Computers & Graphics 31*, 3 (2007), 480–492. 36, 44, 45

[LDW97] Lounsbery M., DeRose T. D., Warren J.: Multiresolution analysis for surfaces of arbitrary topological type. *ACM Transactions on Graphics 16*, 1 (1997), 34–73. 4, 24, 68, 176, 189

[LGD*06] Lavoué G., Gelasca E. D., Dupont F., Baskurt A., Ebrahimi T.: Perceptually driven 3D distance metrics with application to watermarking. In *Proceedings of the SPIE Electronic Imaging* (2006), vol. 6312, pp. 63120L.1–63120L.12. 84, 152, 157, 202

[LK07] Lee S. H., Kwon K. R.: A watermarking for 3D mesh using the patch CEGIs. *Digital Signal Processing 17*, 2 (2007), 396–413. 32, 46, 49, 185

[LK08] Lee S. H., Kwon K. R.: Mesh watermarking based projection onto two convex sets. *Multimedia Systems 13*, 5-6 (2008), 323–330. 31

[LL01] Lu C. S., Liao H. Y. M.: Multipurpose watermarking for image authentication and protection. *IEEE Transactions on Image Processing 10*, 10 (2001), 1579–1592. 67

[LLC00] Lu C. S., Liao H. Y. M., Chen L. H.: Multipurpose audio watermarking. In *Proceedings of the International Conference on Pattern Recognition* (2000), vol. 3, pp. 282–285. 67

[LLKL05] Lee J. W., Lee S. H., Kwon K. R., Lee K. I.: Complex EGI based 3D-mesh watermarking. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences E88*, 6 (2005), 1512–1519. 46

[LLLL05] Lin H. S., Liao H. M., Lu C., Lin J.: Fragile watermarking for authenticating 3-D polygonal meshes. *IEEE Transactions on Multimedia 7*, 6 (2005), 997–1006. 21, 22, 25, 82, 183

[LPG08] Liu Y., Prabhakaran B., Guo X.: A robust spectral approach for blind watermarking of manifold surfaces. In *Proceedings of the ACM Workshop on Multimedia and Security* (2008), pp. 43–52. 36, 37, 46, 49, 131, 132, 139, 144, 163, 185, 196, 198, 200

[LS05] Liu W., Sun S. H.: A robust and invisible watermarking of 3D triangle meshes. In *Proceedings of the International Conference on Knowledge-Based Intelligent Information and Engineering Systems* (2005), pp. 881–888. 31

[LT98] Lindstrom P., Turk G.: Fast and memory efficient polygonal simplification. In *Proceedings of the IEEE Visualization* (1998), pp. 279–286. 155, 203

[LWBL09] Luo M., Wang K., Bors A. G., Lavoué G.: Local patch blind spectral watermarking method for 3D graphics. In *Proceedings of the International Workshop on Digital Watermarking* (2009), pp. 211–226. (to appear). 36, 45, 49, 131, 132, 185

[LZP*04] Li L., Zhang D., Pan Z., Shi J., Zhou K., Ye K.: Watermarking 3D mesh by spherical parameterization. *Computers & Graphics 28*, 6 (2004), 981–989. 40, 49, 185

[LZSCO09] Liu R., Zhang H., Shamir A., Cohen-Or D.: A part-aware surface metric for shape analysis. *Computer Graphics Forum 28*, 2 (2009). (to appear). 169, 209

[MB99] Mintzer F., Braudaway G. W.: If one watermark is good, are more better? In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (1999), pp. 2067–2069. 67, 68

[MDS97] M. D. Swanson B. Zhu A. T.: Data hiding for video-in-video. In *Proceedings of the IEEE International Conference on Image Processing* (1997), vol. 2, pp. 676–679. 52

[ME04] Maret Y., Ebrahimi T.: Data hiding on 3D polygonal meshes. In *Proceedings of the ACM Multimedia and Security Workshop* (2004), pp. 68–74. 31

[MS03] Murotani K., Sugihara K.: Watermarking 3D polygonal meshes using the singular spectrum analysis. In *Proceedings of the IMA International Conference on the Mathematics of Surfaces* (2003), pp. 85–98. 38, 43, 46

[MvOV01] Menezes A. J., van Oorschot P. C., Vanstone S. A.: *Handbook of Applied Cryptography*. CRC Press, 2001. 11, 179

[NF89] Nielson G. M., Foley T. A.: *Mathematical Methods in Computer Aided Geometric Design*. Academic Press, 1989, ch. A survey of applications of an affine invariant norm, pp. 445–467. 43

[NK04] Novotni M., Klein R.: Shape retrieval using 3D Zernike descriptors. *Computer-Aided Design 36*, 11 (2004), 1047–1062. 169, 209

[OM01] Ohbuchi R., Masuda H.: Managing CAD data as a multimedia data type using digital watermarking. In *Proceedings of the IFIP TC5 WG5.2 Workshop on Knowledge Intensive CAD to Knowledge Intensive Engineering* (2001), pp. 103–116. 68

[OMA97] Ohbuchi R., Masuda H., Aono M.: Watermarking three-dimensional polygonal models. In *Proceedings of the ACM Multimedia* (1997), pp. 261–272. xiii, 2, 22, 23, 41, 49, 174, 185

[OMA98] Ohbuchi R., Masuda H., Aono M.: Data embedding algorithms for geometrical and non-geometrical targets in three-dimensional polygonal models. *Computer Communications 21*, 15 (1998), 1344–1354. 31, 32

[OMT02] Ohbuchi R., Mukaiyama A., Takahashi S.: A frequency-domain approach to watermarking 3D shapes. *Computer Graphics Forum 21*, 3 (2002), 373–382. 34, 36, 43, 44, 45, 46, 47, 49, 185

[OMT04] Ohbuchi R., Mukaiyama A., Takahashi S.: Watermarking a 3D shape model defined as a point set. In *Proceedings of the IEEE International Conference on Cyberworlds* (2004), pp. 392–399. 35

[OP98] ÓRuanaidh J. J., Pun T.: Rotation, scale and translation invariant spread spectrum digital image watermarking. *Signal Processing 66*, 3 (1998), 303–317. 52

[OTMM01] Ohbuchi R., Takahashi S., Miyazawa T., Mukaiyama A.: Watermarking 3D polygonal meshes in the mesh spectral domain. In *Proceedings of the Graphics Interface* (2001), pp. 9–17. 43, 44, 45

[PA06] Payan F., Antonini M.: Mean square error approximation for wavelet-based semiregular mesh compression. *IEEE Transactions on Visualization and Computer Graphics 12*, 4 (2006), 649–657. 67

[PAK98] Petitcolas F. A. P., Anderson R. J., Kuhn M. G.: Attacks on copyright marking systems. In *Proceedings of the International Workshop on Information Hiding* (1998), pp. 218–238. 149, 156

[PBBC97] Piva A., Barni M., Bartolini F., Cappellini V.: DCT-based watermark recovering without resorting to the uncorrupted original image. In *Proceedings of the IEEE International Conference on Image Processing* (1997), vol. 1, pp. 520–523. 52, 53

[Pet00] Petitcolas F. A. P.: Watermarking schemes evaluation. *IEEE Signal Processing 17*, 5 (2000), 58–64. 149

[PFCPG05] Pérez-Freire L., Comesaña P., Pérez-González F.: Information-theoretic analysis of security in side-informed data hiding. In *Proceedings of the International Workshop on Information Hiding* (2005), pp. 131–145. 63, 71, 109, 144

[PFCTPPG06] Pérez-Freire L., Comesaña P., Troncoso-Pastoriza J. R., Pérez-González F.: Watermarking security: A survey. *LNCS Transactions on Data Hiding and Multimedia Security 1* (2006), 41–72. 62

[PGMBA05] Pérez-González F., Mosquera C., Barni M., Abrardo A.: Rational dither modulation: A high-rate data-hiding method invariant to gain attacks. *IEEE Transactions on Signal Processing 53*, 10 (2005), 3960–3975. 108

[PH03] Praun E., Hoppe H.: Spherical parametrization and remeshing. In *Proceedings of the ACM Siggraph* (2003), pp. 340–349. 40

[PHF99] Praun E., Hoppe H., Finkelstein A.: Robust mesh watermarking. In *Proceedings of the ACM Siggraph* (1999), pp. 49–56. 39, 45, 49, 185

[PVM*01] Pereira S., Voloshynovskiy S., Madueno M., Marchand-Maillet S., Pun T.: Second generation benchmarking and application oriented evaluation. In *Proceedings of the International Workshop on Information Hiding* (2001), pp. 340–353. 149, 168, 209

[RA99] Ramkumar M., Akansu A.: Self-noise suppression schemes in blind image steganography. In *Proceedings of the SPIE Electronic Imaging* (1999), vol. 3845, pp. 55–66. 55, 60

[RAM05] Rondao-Alface P., Macq B.: Blind watermarking of 3D meshes using robust feature points detection. In *Proceedings of the IEEE International Conference on Image Processing* (2005), vol. 1, pp. 693–696. 49, 73, 132, 185

[RAM06] Rondao-Alface P., Macq B.: Shape quality measurement for 3D watermarking schemes. In *Proceedings of the SPIE Electronic Imaging* (2006), vol. 6072, pp. 622–634. 84

[RAM07] Rondao-Alface P., Macq B.: From 3D mesh data hiding to 3D shape blind and robust watermarking: A survey. *LNCS Transactions on Data Hiding and Multimedia Security 2* (2007), 99–115. 18

[RAMC07] Rondao-Alface P., Macq B., Cayre F.: Blind and robust watermarking of 3D models: How to withstand the cropping attack? In *Proceedings of the IEEE International Conference on Image Processing* (2007), vol. 5, pp. 465–468. 45, 102, 132

[RCB05] Ricard J., Coeurjolly D., Baskurt A.: Generalizations of angular radial transform for 2D and 3D shape retrieval. *Pattern Recognition Letters 26*, 14 (2005), 2174–2186. 169, 209

[Rus07] Rustamov R. M.: Laplace-Beltrami eigenfunctions for deformation invariant shape representation. In *Proceedings of the Symposium on Geometry processing* (2007), pp. 225–233. 170, 210

[SC04] Song H. S., Cho N. I.: Digital watermarking of 3D geometry. In *Proceedings of the International Symposium on Intelligent Signal Processing and Communication Systems* (2004), pp. 272–277. 33

[Sch64] Schuchman L.: Dither signals and their effect on quantization noise. *IEEE Transactions on Communication Technology 12*, 4 (1964), 162–165. 58, 60

[SCOT03] Sorkine O., Cohen-Or D., Toledo S.: High-pass quantization for mesh encoding. In *Proceedings of the Symposium on Geometry Processing* (2003), pp. 42–51. 44, 98, 104, 191

[SK01] Sequeira A., Kundur D.: Communication and information theory in watermarking: A survey. In *Proceedings of the SPIE Electronic Imaging* (2001), vol. 4518, pp. 216–227. 54

[SRA06] Sencar H. T., Ramkumar M., Akansu A. N.: An overview of scalar quantization based data hiding methods. *Signal Processing 86*, 5 (2006), 893–914. xvii, 54, 55

[SS95] Schröder P., Sweldens W.: Spherical wavelets: Efficiently representing functions on the sphere. In *Proceedings of the ACM Siggraph* (1995), pp. 161–172. 38

[SSCO08] Shapira L., Shamir A., Cohen-Or D.: Consistent mesh partitioning and skeletonization using the shape diameter function. *The Visual Computer 24*, 4 (2008), 249–259. 169, 209

[SSNO01] Sheppard N. P., Safavi-Naini R., Ogunbona P.: On multiple watermarking. In *Proceedings of the International Workshop on Multimedia and Security* (2001), pp. 3–6. 67, 68

[ST01] Sheynin S. A., Tuzikov A. V.: Explicit formulae for polyhedra moments. *Pattern Recognition Letters 22*, 10 (2001), 1103–1109. 99

[STN*01] Solachidis V., Tefas A., Nikolaidis N., Tsekeridou S., Nikolaidis A., Pitas I.: A benchmarking protocol for watermarking methods. In *Proceedings of the IEEE International Conference on Image Processing* (2001), vol. 3, pp. 1023–1026. 149

[Tau00] Taubin G.: Geometric signal processing on polygonal meshes. In *Proceedings of the Eurographics State-of-the-art Reports* (2000), pp. 81–96. 87, 154

[TCR03] Toledo S., Chen D., Rotkin V.: Taucs: A library of sparse linear solvers, 2003. Software available at `http://www.tau.ac.il/~stoledo/taucs/` (access date: 14 July 2009). 136, 197

[TNM90] Tanaka K., Nakamura Y., Matsui K.: Embedding secret information into a dithered multi-level image. In *Proceedings of the IEEE Military Communications Conference* (1990), vol. 1, pp. 216–220. 52

[TSV03] Tuzikov A. V., Sheynin S. A., Vasiliev P. V.: Computation of volume and surface body moments. *Pattern Recognition 36*, 11 (2003), 2521–2529. 99, 102, 154, 192

[TWC*06] Tsai Y. Y., Wang C. M., Cheng Y. M., Chang C. H., Wang P. C.: Steganography on 3D models using a spatial subdivision technique. In *Proceedings of the Computer Graphics International* (2006), pp. 469–476. 26, 28, 184

[UCB04] Uccheddu F., Corsini M., Barni M.: Wavelet-based blind watermarking of 3D models. In *Proceedings of the ACM Multimedia and Security Workshop* (2004), pp. 143–154. 38, 49, 185

[VL07] Vallet B., Lévy B.: *Manifold Harmonics*. Tech. rep., INRIA - ALICE Project Team, 2007. xiv, 37, 131, 132, 134, 136, 196, 197

[VL08] Vallet B., Lévy B.: Spectral geometry processing with manifold harmonics. *Computer Graphics Forum 27*, 2 (2008), 251–260. 4, 37, 131, 132, 136, 176, 196, 197

[VP04] Valette S., Prost R.: Wavelet-based multiresolution analysis of irregular surface meshes. *IEEE Transactions on Visualization and Computer Graphics 10*, 2 (2004), 113–122. 39

[VPP*01] Voloshynovskiy S., Pereira S., Pun T., Eggers J. J., Su J. K.: Attacks on digital watermarks: Classification, estimation-based attacks, and benchmarks. *IEEE Communications Magazine 39* (2001), 118–126. 168, 209

[vSTO94] van Schyndel R. G., Tirkel A., Osborne C.: A digital watermark. In *Proceedings of the IEEE International Conference on Image Processing* (1994), vol. 2, pp. 86–90. 52

[Wag00] Wagner M. G.: Robust watermarking of polygonal meshes. In *Proceedings of the Geometric Modeling and Processing* (2000), pp. 201–208. 43

[WBSS04] Wang Z., Bovik A., Sheikh H., Simoncelli E.: Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing 13*, 4 (2004), 1–14. 152

[WC05] Wang C. M., Cheng Y. M.: An efficient information hiding algorithm for polygon models. *Computer Graphics Forum 24*, 3 (2005), 591–600. 26, 28, 41, 184

[WC06] Wu H. T., Cheung Y. M.: A high-capacity data hiding method for polygonal meshes. In *Proceedings of the International Workshop on Information Hiding* (2006), pp. 188–200. 22, 25, 183

[WD96] Wolfgang R. B., Delp E. J.: A watermark for digital images. In *Proceedings of the IEEE International Conference on Image Processing* (1996), vol. 3, pp. 219–222. 52

[WK05] Wu J., Kobbelt L. P.: Efficient spectral watermarking of large meshes with orthogonal basis functions. *The Visual Computer 21*, 8–10 (2005), 848–857. 38, 43, 44, 46, 47, 49, 185

[WLBD09] Wang K., Luo M., Bors A. G., Denis F.: Blind and robust mesh watermarking using manifold harmonics. In *Proceedings of the IEEE International Conference on Image Processing* (2009). (to appear). 130

[WLDB07a] Wang K., Lavoué G., Denis F., Baskurt A.: Hierarchical blind watermarking of 3D triangular meshes. In *Proceedings of the IEEE International Conference on Multimedia & Expo* (2007), pp. 1235–1238. 66

[WLDB07b] Wang K., Lavoué G., Denis F., Baskurt A.: Three-dimensional meshes watermarking: Review and attack-centric investigation. In *Proceedings of the International Workshop on Information Hiding* (2007), pp. 50–64. 18

[WLDB08a] Wang K., Lavoué G., Denis F., Baskurt A.: A comprehensive survey on three-dimensional mesh watermarking. *IEEE Transactions on Multimedia 10*, 8 (2008), 1513–1527. 18, 97

[WLDB08b] Wang K., Lavoué G., Denis F., Baskurt A.: A fragile watermarking scheme for authentication of semi-regular meshes. In *Proceedings of the Eurographics Short Papers* (2008), pp. 5–8. 66

[WLDB08c] Wang K., Lavoué G., Denis F., Baskurt A.: Hierarchical watermarking of semi-regular meshes based on wavelet transform. *IEEE Transactions on Information Forensics and Security 3*, 4 (2008), 620–634. 66

[WLDB09a] Wang K., Lavoué G., Denis F., Baskurt A.: A benchmark for 3-D mesh watermarking, 2009. (to be submitted) (research report version available on-line at `http://liris.cnrs.fr/Documents/Liris-4143.pdf`). 148

[WLDB09b] Wang K., Lavoué G., Denis F., Baskurt A.: Robust and blind watermarking of polygonal meshes based on volume moments, 2009. Submitted to the *IEEE Transactions on Visualization and Computer Graphics* (under major revisions) (research report version available on-line at `http://liris.cnrs.fr/Documents/Liris-3713.pdf`). 96

[WSBD00] Wood Z. J., Schröder P., Breen D., Desbrun M.: Semi-regular mesh extraction from volumes. In *Proceedings of the IEEE Visualization* (2000), pp. 275–282. 67

[WZYG08] Wang W. B., Zheng G. Q., Yong J. H., Gu H. J.: A numerically stable fragile watermarking scheme for authenticating 3D models. *Computer-Aided Design 40*, 5 (2008), 634–645. 22, 25, 183

[XLP07] Xin Y., Liao S., Pawlak M.: Circularly orthogonal moments for geometrically robust image watermarking. *Pattern Recognition 40*, 12 (2007), 3740–3752. 99

[YIK03] Yu Z., Ip H. H. S., Kwok L. F.: A robust watermarking scheme for 3D triangular mesh models. *Pattern Recognition 36*, 11 (2003), 2603–2614. xiii, 28, 29, 45, 47, 49, 185

[YKL06] Yang J. H., Kim C. S., Lee S. U.: Semi-regular representation and progressive compression of 3-D dynamic mesh sequences. *IEEE Transactions on Image Processing 15*, 9 (2006), 2531–2544. 67

[YPSZ01] Yin K., Pan Z., Shi J., Zhang D.: Robust mesh watermarking based on multiresolution processing. *Computers & Graphics 25*, 3 (2001), 409–420. 39, 44, 45, 47, 49, 185

[YY99] Yeo B., Yeung M. M.: Watermarking 3D objects for verification. *IEEE Computer Graphics and Applications 19*, 1 (1999), 36–45. xiii, 21, 22, 24, 25, 183

[ZC01] Zhang C., Chen T.: Efficient feature extraction for 2D/3D objects in mesh representation. In *Proceedings of the IEEE International Conference on Image Processing* (2001), pp. 935–938. 4, 98, 99, 102, 154, 176, 192

[ZS00] Zorin D., Schröder P.: Subdivision for modeling and animation. In *Proceedings of the ACM Siggraph Course Notes* (2000). 117, 155, 203

[ZTP05] Zafeiriou S., Tefas A., Pitas I.: Blind robust watermarking schemes for copyright protection of 3D mesh objects. *IEEE Transactions on Visualization and Computer Graphics 11*, 5 (2005), 596–607. 30, 31, 44, 46, 49, 97, 102, 169, 185, 209

[ZvKD07] Zhang H., van Kaick O., Dyer R.: Spectral methods for mesh processing and analysis. In *Proceedings of the Eurographics State-of-the-art Report* (2007), pp. 1–22. 44, 98, 104, 191

Author's Publications

International Journals

- A comprehensive survey on three-dimensional mesh watermarking, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, *IEEE Transactions on Multimedia*, vol. 10, no. 8, pp. 1513-1527, 2008.
- Hierarchical watermarking of semiregular meshes based on wavelet transform, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 4, pp. 620-634, 2008.

International Conferences

- Three-dimensional meshes watermarking: Review and attack-centric investigation, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, in *Proceedings of the International Workshop on Information Hiding*, Saint-Malo, France, June 2007, Springer Lecture Notes in Computer Science, vol. 4567, pp. 50-64.
- Hierarchical blind watermarking of 3D triangular meshes, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, in *Proceedings of the IEEE International Conference on Multimedia & Expo*, Beijing, China, July 2007, pp. 1235-1238.
- A fragile watermarking scheme for authentication of semi-regular meshes, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, in *Proceedings of the Eurographics Short Papers*, Crete, Greece, April 2008, pp. 5-8.
- Local patch blind spectral watermarking method for 3D graphics, Ming Luo, **Kai Wang**, Adrian G. Bors, and Guillaume Lavoué, in *Proceedings of the International Workshop on Digital Watermarking*, Surrey, United Kingdom, August 2009, Springer Lecture Notes in Computer Science, vol. 5703, pp. 211-226.
- Blind and robust mesh watermarking using manifold harmonics, **Kai Wang**, Ming Luo, Adrian G. Bors, and Florence Denis, in *Proceedings of the IEEE International Conference on Image Processing*, Cairo, Egypt, November 2009. (to appear)

Book Chapter

- Blind watermarking of three-dimensional meshes: Review, recent advances and future opportunities, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, *as a Chapter in the Edited Book "Advanced Techniques in Multimedia Watermarking: Image, Video and Audio Applications"*, IGI Global, 2009. (to appear)

Local Conferences

- Tatouage hiérarchique et aveugle de maillages tridimensionnels, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, in *Actes des Journées d'Étude et d'Échange COmpression et REprésentation des Signaux Audiovisuels (CORESA'07)*, Montpellier, France, November 2007, pp. 139-143. (in French)
- Tatouage robuste et aveugle de maillages 3D basé sur les moments volumiques, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, in *Actes des Journées d'Étude et d'Échange COmpression et REprésentation des Signaux Audiovisuels (CORESA'09)*, Toulouse, France, March 2009, pp. 162-167. (in French)

Seminars

- État de l'art en tatouage de maillages 3D, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, *Presented at the SISS Project Seminar of the ISLE Cluster of the Rhône-Alpes Region*, Lyon, France, 1 March 2007. (in French)
- Tatouage hiérarchique et aveugle de maillages tridimensionnels, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, *Presented at the GdR ISIS Seminar*, Lyon, France, 11 October 2007. (in French)
- Tatouage robuste et aveugle de maillages 3D en utilisant les harmoniques variétés, **Kai Wang**, Ming Luo, Adrian G. Bors, Florence Denis, and Guillaume Lavoué, *Presented at the GdR ISIS Seminar*, Paris, France, 12 March 2009. (in French)

Research Reports

- Robust and blind watermarking of polygonal meshes based on volume moments, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, Research Report of LIRIS, *Submitted to IEEE Transactions on Visualization and Computer Graphics*, October 2008. (under major revisions)
- A benchmark for 3-D mesh watermarking, **Kai Wang**, Guillaume Lavoué, Florence Denis, and Atilla Baskurt, Research Report of LIRIS, *to be submitted*, October 2009.

Title: Quantization-Based Blind Watermarking of Three-Dimensional Meshes

Abstract: With the increasing use of three-dimensional (3-D) models in various practical applications, more and more attention has been paid on the research of digital watermarking techniques for 3-D polygonal meshes. In this thesis, we first provide a comprehensive survey on the state of the art in 3-D mesh watermarking, with an original attack-centric investigation. Then, we make use of the scalar Costa quantization scheme to construct a number of effective blind mesh watermarking schemes. We successfully embed multi-bit quantization-based blind watermarks in three different mesh domains: the wavelet domain of a semi-regular mesh, and the spatial and spectral domains of a general mesh. The watermarking primitives, which are subject to scalar Costa quantization, are respectively the norms and orientations of the wavelet coefficient vectors, the analytic volume moments and the manifold harmonics spectral amplitudes. Finally, we detail the design and implementation of a robust mesh watermark benchmarking system, which has been made publicly available on-line. This benchmarking system comprises a standard mesh data set, a software tool and two application-oriented evaluation protocols. The robust mesh watermarking schemes proposed in this thesis and a state-of-the-art method are compared within this benchmarking framework. The comparison results demonstrate both the effectiveness of our blind watermarking schemes and the relevance of our benchmarking system.

Keywords: 3-D mesh, blind watermarking, scalar Costa scheme, wavelet, volume moment, manifold harmonics, benchmark.

Titre : Tatouage Aveugle de Maillages 3D Basé sur la Quantification

Résumé : Avec l'accroissement de l'utilisation des objets tridimensionnels (3D) dans diverses applications, une attention de plus en plus forte a été portée sur la protection de ce contenu 3D par des techniques de tatouage numérique. Dans ce travail de thèse, nous avons réalisé d'abord un état de l'art complet sur le domaine du tatouage de maillages, avec un point de vue original centré sur les attaques. Puis, nous avons proposé plusieurs méthodes de tatouage aveugles basées sur des techniques de quantification par le schéma de Costa scalaire (SCS). Nous avons choisi différents domaines appropriés d'un maillage 3D pour l'insertion de tatouages aveugles : le domaine ondelettes d'un maillage semi-régulier, et les domaines spatial et spectral d'un maillage général. Les primitives de tatouage, qui sont soumises à la quantification scalaire, sont respectivement les normes et les orientations des vecteurs de coefficients d'ondelettes, les moments volumiques analytiques et les amplitudes spectrales en harmoniques variétés. Enfin, nous avons conçu et implémenté un système de benchmark pour les techniques de tatouage robustes de maillages. Ce benchmark est accessible librement sur Internet et contient une collection de modèles 3D, un outil logiciel et deux protocoles d'évaluation orientés sur différentes applications. Les méthodes de tatouage robustes proposées dans ce manuscrit ainsi qu'une méthode récente de l'état de l'art sont comparées grâce à ce benchmark. Les résultats obtenus montrent l'efficacité des méthodes proposées ainsi que la pertinence du système de benchmark.

Mots clés : Maillage 3D, tatouage aveugle, schéma de Costa scalaire, ondelettes, moment volumique, harmoniques variétés, benchmark.

FOLIO ADMINISTRATIF

THESE SOUTENUE DEVANT L'INSTITUT NATIONAL DES SCIENCES APPLIQUEES DE LYON

NOM : WANG
(avec précision du nom de jeune fille, le cas échéant)

DATE de SOUTENANCE : 6 Novembre 2009

Prénoms : Kai

TITRE : Tatouage Aveugle de Maillages 3D Basé sur la Quantification

NATURE : Doctorat

Numéro d'ordre : 2009-ISAL-0082

Ecole doctorale : Ecole Doctorale Informatique et Mathématiques (InfoMaths)

Spécialité : Informatique

Cote B.I.U. - Lyon : T 50/210/19 / et bis

CLASSE :

RESUME :
Avec l'accroissement de l'utilisation des objets tridimensionnels (3D) dans diverses applications, une attention de plus en plus forte a été portée sur la protection de ce contenu 3D par des techniques de tatouage numérique. Dans ce travail de thèse, nous avons réalisé d'abord un état de l'art complet sur le domaine du tatouage de maillages, avec un point de vue original centré sur les attaques. Puis, nous avons proposé plusieurs méthodes de tatouage aveugles basées sur des techniques de quantification par le schéma de Costa scalaire (SCS). Nous avons choisi différents domaines appropriés d'un maillage 3D pour l'insertion de tatouages aveugles : le domaine ondelettes d'un maillage semi-régulier, et les domaines spatial et spectral d'un maillage général. Les primitives de tatouage, qui sont soumises à la quantification scalaire, sont respectivement les normes et les orientations des vecteurs de coefficients d'ondelettes, les moments volumiques analytiques et les amplitudes spectrales en harmoniques variétés. Enfin, nous avons conçu et implémenté un système de benchmark pour les techniques de tatouage robustes de maillages. Ce benchmark est accessible librement sur Internet et contient une collection de modèles 3D, un outil logiciel et deux protocoles d'évaluation orientés sur différentes applications. Les méthodes de tatouage robustes proposées dans ce manuscrit ainsi qu'une méthode récente de l'état de l'art sont comparées grâce à ce benchmark. Les résultats obtenus montrent l'efficacité des méthodes proposées ainsi que la pertinence du système de benchmark.

MOTS-CLES :
Maillage 3D, tatouage aveugle, schéma de Costa scalaire, ondelettes, moment volumique, harmoniques variétés, benchmark.

Laboratoire (s) de recherche : Laboratoire d'InfoRmatique en Image et Systèmes d'information (LIRIS)

Directeur de thèse: Pr. Atilla Baskurt

Président de jury :

Composition du jury :
M. Mauro Barni, Professeur, Università di Siena (Rapporteur)
M. Bruno Lévy, Directeur de Recherche, INRIA Nancy Grand Est (Rapporteur)
M. Benoît Macq, Professeur, Université Catholique de Louvain (Rapporteur)
M. Jean-Marc Chassery, Directeur de Recherche CNRS, GIPSA-lab Grenoble (Examinateur)
Mme. Caroline Fontaine, Chargée de Recherche CNRS, IRISA Rennes (Examinateur)
M. Atilla Baskurt, Professeur, INSA Lyon (Directeur)
Mme. Florence Denis, Maître de Conférences, Université Lyon 1 (Co-encadrant)
M. Guillaume Lavoué, Maître de Conférences, INSA Lyon (Co-encadrant)