N° d'ordre Année 2007

THESE

présentée devant

L'INSTITUT NATIONAL DE SCIENCES APPLIQUEES DE LYON
ET
YEUNGNAM UNIVERSITY

pour obtenir

LE GRADE DE DOCTEUR

FORMATION DOCTORALE : SIDS spécialité ISSI
ECOLE DOCTORALE : ELECTRONIQUE, ELECTROTECHNIQUE, AUTOMATIQUE

PAR

MIN-SU KIM

Wavelet Transform Based Digital Watermarking for 3-D Surface Meshes and Mesh Sequences

Soutenue le 26 février 2007 devant la Commission d'Examen

Jury :

Francis SCHMITT	Rapporteur
Ki-Ryong KWON	Rapporteur
Isabelle MAGNIN	Examinateur
Kook-Yeol YOO	Examinateur
Rémy PROST	Co-Directeur de thèse
Ho-Youl JUNG	Co-Directeur de thèse

Cette thèse a été préparée au,
Centre de Recherche Et d'Applications en Traitement de l'Image et du Signal (CREATIS UMR CNRS 5515) de l'INSA LYON et l'UCBL
Multimedia Signal Processing (MSP) Laboratory of Yeungnam University

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
	CHIMIE DE LYON **M. Denis SINOU**	**M. Denis SINOU** Université Claude Bernard Lyon 1 Lab Synthèse Asymétrique UMR UCB/CNRS 5622 Bât 308, 2ème étage 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cedex Tél : 04.72.44.81.83 Fax : 04.78.89.89.14 sinou@univ-lyon1.fr
E2MC	ECONOMIE, ESPACE ET MODELISATION DES COMPORTEMENTS **M. Alain BONNAFOUS**	**M. Alain BONNAFOUS** Université Lyon 2 14 avenue Berthelot MRASH M. Alain BONNAFOUS Laboratoire d'Economie des Transports 69363 LYON Cedex 07 Tél : 04.78.69.72.76 Alain.bonnafous@ish-lyon.cnrs.fr
E.E.A.	ELECTRONIQUE, ELECTROTECHNIQUE, AUTOMATIQUE **M. Daniel BARBIER**	**M. Daniel BARBIER** INSA DE LYON Laboratoire Physique de la Matière Bâtiment Blaise Pascal 69621 VILLEURBANNE Cedex Tél : 04.72.43.64.43 Fax : 04.72.43.60.82 Daniel.Barbier@insa-lyon.fr
E2M2	EVOLUTION, ECOSYSTEME, MICROBIOLOGIE, MODELISATION http://biomserv.univ-lyon1.fr/E2M2 **M. Jean-Pierre FLANDROIS**	**M. Jean-Pierre FLANDROIS** UMR 5558 Biométrie et Biologie Evolutive Equipe Dynamique des Populations Bactériennes Faculté de Médecine Lyon-Sud Laboratoire de Bactériologie BP 1269600 OULLINS Tél : 04.78.86.31.50 Fax : 04.72.43.13.88 E2m2@biomserv.univ-lyon1.fr
EDIIS	INFORMATIQUE ET INFORMATION POUR LA SOCIETE http://www.insa-lyon.fr/ediis **M. Lionel BRUNIE**	**M. Lionel BRUNIE** INSA DE LYON EDIIS, Bâtiment Blaise Pascal 69621 VILLEURBANNE Cedex Tél : 04.72.43.60.55 Fax : 04.72.43.60.71 ediis@insa-lyon.fr
EDISS	INTERDISCIPLINAIRE SCIENCES-SANTE http://www.ibcp.fr/ediss **M. Alain Jean COZZONE**	**M. Alain Jean COZZONE** IBCP (UCBL1) 7 passage du Vercors 69367 LYON Cedex 07 Tél : 04.72.72.26.75 Fax : 04.72.72.26.01 cozzone@ibcp.fr
	MATERIAUX DE LYON **M. Jacques JOSEPH**	**M. Jacques JOSEPH** Ecole Centrale de Lyon Bât F7 Lab. Sciences et Techniques des Matériaux et des Surfaces 36 Avenue Guy de Collongue BP 163 69131 ECULLY Cedex Tél : 04.72.18.62.51 Fax : 04.72.18.60.90 Jacques.Joseph@ec-lyon.fr
Math IF	MATHEMATIQUES ET INFORMATIQUE FONDAMENTALE http://www.ens-lyon.fr/MathIS **M. Franck WAGNER**	**M. Franck WAGNER** Université Claude Bernard Lyon1 Institut Girard Desargues UMR 5028 MATHEMATIQUES Bâtiment Doyen Jean Braconnier Bureau 101 Bis, 1er étage 69622 VILLEURBANNE Cedex Tél : 04.72.43.27.86 Fax : 04.72.43.16.87
MEGA	MECANIQUE, ENERGETIQUE, GENIE CIVIL, ACOUSTIQUE http://www.lmfa.ec-lyon.fr/autres/MEGA/index.html **M. François SIDOROFF**	**M. François SIDOROFF** Ecole Centrale de Lyon Lab. Tribologie et Dynamique des Systêmes Bât G8, 36 avenue Guy de Collongue, BP 163 69131 ECULLY Cedex Tél : 04.72.18.62.14 Fax : 04.72.18.65.37 Francois.Sidoroff@ec-lyon.fr

Abstract

This thesis deals with digital watermarking methods for copyright protection of 3-D surface meshes and mesh sequences. The proposed methods are based on wavelet analysis of the geometry of the surface meshes, or on wavelet analysis of the vertex coordinate signals along the time axis.

3-D static meshes

The first proposal embeds the watermark into the L2 norm of the geometric wavelet coefficients (vectors) by using spread spectrum method. The Cartesian coordinates should be converted into the spherical coordinates. The method can directly process semi-regular or irregular meshes. For the invariance of the wavelet coefficients after inverse wavelet transform, it is necessary to synchronize the connectivity by re-ordering the vertex indexes from a reference vertex. This approach ensures the blind detection which does not require the original meshes during watermark extraction procedures.

The second proposal is based on a statistical approach. The histogram of the L2 norm of scale coefficients (approximation mesh) was divided into regular bins. The mean (or the variance) of each bin is modified according to the watermark by a non-linear exponential transformation. The watermark can be extracted from the distribution of the scale coefficients after wavelet analysis, as well as from the distribution of vertex norms of the watermarked meshes without using the wavelet analysis. This proposal is more robust against topological attacks than the first one.

3-D mesh sequence (3-D dynamic meshes with fixed connectivity, 3D+t)

The first proposal is based on the vertex coordinate signals along the time axis, and the second uses the vertex norm signals along the time axis of the mesh sequence. As a result, the number of signals to decompose by wavelet analysis equals three times of the number of vertices for the first proposal and equals the number of vertices for the second.

The first proposal embeds the watermark into the temporal wavelet (high frequency) coefficients of each coordinates of the sequence. In this proposal, the histogram of L2 norm of the scale (low frequency) coefficients is divided into distinct bins with equal range according to their magnitude. Since the distribution of temporal wavelet coefficients can be approximated to Laplacian distribution, the variance of the distribution of each bin is asymmetrically modified according to the watermark bits to be embedded. This method is robust against frame-averaging attack and frame-by-frame attacks.

In the second proposal, the Cartesian coordinates of the vertices are converted into the spherical coordinates and the L2 norm of each vertex is wavelet transformed along the time axis. Similar to the first proposal, the variance of the distribution of wavelet coefficients is modified according to the watermark. This proposal which uses L2 norm outperforms the first proposal in terms of the invisibility of the watermark and the robust against rotation attacks.

Finally, we propose the method which embeds the watermark into both wavelet and scale coefficients. This method allows extracting the watermark from the distribution of scale and wavelet coefficients of each frame. In addition, it is possible to extract the watermark from a single frame in spatial domain.

Note that all of our proposals do not require the mesh sequence (or original meshes) in the procedures of watermark extraction.

The robustness against geometrical attacks (additive noise on the vertex coordinates, uniform vertex coordinates quantization, low-pass filtering, cropping, rotation, translation, scaling, frame-dropping for mesh sequence), and topological attacks (random vertex index reordering, simplification, subdivision) are evaluated on the semi-regular and irregular meshes.

Keywords

3-D Surface Meshes, 3-D Mesh Sequence, Digital Watermarking, Copyright Protection, Wavelet, Blind Detection

Résumé

Cette thèse propose des méthodes de protection des droits d'auteur, par tatouages numériques, relatifs à des objets en trois dimensions représentés par leurs maillages triangulaires surfaciques statiques ou dynamiques. Les approches proposées utilisent la décomposition en ondelettes du maillage ou des signaux associés à l'évolution des coordonnées des sommets de ce maillage au cours du temps.

Maillages statiques 3D

Une première proposition, inspirée des méthodes dite 'spread spectrum', incorpore le tatouage dans le module des coefficients (vecteurs) d'ondelettes géométriques. Le maillage peut être semi régulier ou irrégulier. Elle nécessite la transformation des coordonnées cartésiennes des sommets en coordonnées sphériques. Après la transformée en ondelettes inverse, les index des sommets sont réordonnés, à partir d'un sommet de référence, afin d'assurer une invariance des coefficients d'ondelettes, nécessaire à l'extraction du tatouage. Cette approche garantit l'extraction du tatouage sans le maillage original.

Une deuxième proposition utilise une approche statistique. Elle incorpore le tatouage dans le module des coefficients d'échelles (maillage d'approximation), par intervalles de leur histogramme. La moyenne (ou la variance) de chaque intervalle est modifiée par le tatouage à l'aide d'une transformation non linéaire exponentielle. Le tatouage peut être extrait, sans le maillage original, à partir de la distribution des coefficients d'échelles ou directement sur le maillage en pleine résolution (domaine spatial). Cette deuxième proposition est plus robuste aux attaques topologiques que la précédente.

Séquence de maillages 3D (Maillages dynamiques à connectivité constante, 3D+t)

Dans une première approche, les évolutions temporelles des coordonnées des sommets sont assimilées à trois signaux indépendants et, dans une deuxième approche, le module de chaque sommet est un signal. Ainsi, le nombre de signaux à décomposer sur une base d'ondelettes est trois fois le nombre de sommets pour la première approche et seulement égal au nombre de sommets dans la seconde.

Pour la première approche le tatouage est incorporé dans les coefficients d'ondelettes temporels de chaque coordonnée de la séquence. Les sommets du maillage sont classés dans des intervalles réguliers de l'histogramme des moyennes temporelles du module des coefficients d'échelles (basses fréquences). Le tatouage modifie la distribution Laplacienne des coefficients d'ondelettes (hautes fréquences) de chacune des classes, de façon asymétrique par une transformation non linéaire exponentielle. Dans chaque intervalle, la variance est modifiée par le tatouage. Cette approche est robuste face aux simplifications par moyenne de trames et aux attaques individuelles de chacune des trames.

Pour la deuxième approche les coordonnées cartésiennes des sommets sont transformées en coordonnées sphériques et le module de chaque sommet est un signal décomposé en ondelettes. Le tatouage est incorporé sur les coefficients d'ondelettes des modules de façon analogue à la méthode précédente. Une variante de cette méthode incorpore, en plus, le tatouage dans les coefficients d'échelles. Elle permet l'extraction du tatouage à l'aide des coefficients d'échelles et d'ondelettes temporels de l'ensemble des trames ou à l'aide d'une seule trame dans le domaine spatial.

On notera que, aussi bien pour les maillages dynamiques que statiques, toutes les méthodes proposées permettent l'extraction du tatouage sans le maillage original (extraction aveugle).

La robustesse, face à des attaques géométriques (addition de bruit sur les coordonnées des sommets, quantification des coordonnées des sommets, filtrage géométrique passe-bas, sélection d'une partie du maillage, rotations, translations, changements d'échelle, suppression de trame pour les maillages dynamiques) et topologiques (réordonnancement des index des sommets, simplification, subdivision) est évaluée sur des maillages semi réguliers et irréguliers.

Mots clés

Maillage surfacique 3D, séquence de maillages 3D, tatouage numérique, protection de 'copyright', ondelette, extraction aveugle.

터마크를 삽입한다.

제안된 첫번째 방법은 메쉬 시퀀스의 각 꼭지점 좌표 축의 시간축 웨이블릿 변환 후, 고주파수 대역에 워터마크를 삽입한다. 제안된 방법은 워터마크의 강인성(robustness) 및 비지각성(invisibility)을 보장하기 위해서 저주파수 대역에 속하는 웨이블릿 계수 벡터 크기값의 확률분포를 일정한 구간으로 나누고 이에 대응되는 고주파수 대역 신호를 수정한다. 시간축 웨이블릿 계수의 확률분포는 라플라시안(Laplacian)으로 근사화될 수 있으며, 제안된 방법은 삽입할 워터마크에 따라 각 구간의 확률분포를 평균을 기준으로 비대칭이 되도록 수정한다. 제안된 방법은 프레임 평균 공격(frame averaing) 뿐 아니라 단일 프레임 공격(frame-by-frame attack)에도 강인함을 보였다.

제안된 두번째 방법은 꼭지점 좌표를 구면 좌표계로 변환한 후 좌표 벡터의 크기값을 시간축 웨이블릿 변환하고, 첫번째 방법과 마찬가지로 웨이블릿 계수의 분산값을 수정함으로써 워터마크를 삽입한다. 제안된 방법은 워터마크의 비가시성 및 회전 공격에 대한 강인성에 있어 첫번째 방법보다 우수하다. 제안된 두번째 방법은 고주파수 대역 뿐만 아니라 저주파수 대역에도 프레임 단위로 워터마크를 삽입하는 방식으로 확장되었다. 이 때, 워터마크는 시간축 웨이블릿 변환 후, 고주파수 대역에 속하는 웨이블릿 계수 및 저주파수 대역에 속하는 프레임단위로 독립적으로 워터마크를 검출할 수 있다. 또한, 저주파수 대역 프레임에 삽입된 워터마크는 모든 프레임으로 확산되어 존재하기 때문에 시간축 웨이블릿 변환과정 없이 단일 프레임에서 워터마크 검출이 가능하다.

본 논문에서 제안한 모든 방법들은 워터마크 검출 시 원본 메쉬 (또는 원본 메쉬 시퀀스)를 사용하지 않는 블라인드 검출 기법을 사용하였다.

또한 제안된 방법들의 성능을 분석하기 위해 반정규 및 비정규 정지 메쉬 및 동형 메쉬 시퀀스에 워터마크를 삽입한 후 객관적인 화질 측정을 통해 비지각성을 평가하였고, 다양한 기하 공격(부가 잡음(adding noise), 꼭지점 좌표값의 균일 양자화(uniform quantization), 저대역 통과 필터(low-pass filtering), 자르기(cropping), 회전(rotation), 이동(translation), 균일 스케일링(uniform scaling), 메쉬 시퀀스의 프레임 제거(frame dropping)) 및 위상 공격(랜덤 꼭지점 인덱스 재배열(vertex re-ordering), 간략화(simplification), 분할(sub-division))을 수행하여 제안된 방법들의 강인성을 평가하였다.

키워드

3차원 표면 메쉬 영상, 3차원 메쉬 시퀀스, 디지털 워터마킹, 저작권 보호, 웨이블릿, 블라인드 검출

요약문

본 학위논문은 3차원 정지 메쉬(3-D static mesh) 및 동형 메쉬 시퀀스(isomorphic mesh sequence)의 저작권 보호(copyright protection)를 위한 웨이블릿 변환(wavelet transform) 기반 디지털 워터마킹(digital watermarking) 기법을 제안한다. 제안된 방법은 정지 메쉬 및 메쉬 시퀀스의 기하정보(geometry information)의 웨이블릿 분해 기법 및 꼭지점 좌표값(vertex coordinate)의 시간축 웨이블릿 분해 기법에 기반한 방법이다.

3차원 정지 메쉬 워터마킹

본 논문에서 제안된 첫번째 방법은 대역 확산(spread spectrum) 기법을 이용하여 웨이블릿 계수 벡터의 크기값(wavelet coefficient norms)에 워터마크(watermark)를 삽입한다. 직교좌표계(Cartesian coordinates)로 표현되는 웨이블릿 계수는 구면 좌표계(spherical coordinates)로 변환함으로써 벡터 크기값을 구할 수 있다. 제안된 방법은 비정규(irregular) 웨이블릿 변환기법을 이용하기 때문에 반정규(semi-regular) 및 비정규 메쉬를 리메쉬(re-meshing)와 같은 전처리과정 없이 워터마크를 삽입할 수 있다. 비정규 웨이블릿 변환 후에 발생하는 꼭지점 인덱스의 동기화(synchronization) 문제를 해결하기 위해, 워터마크 삽입 및 검출(watermark embedding and extraction) 시 웨이블릿 변환 이전에 기준 꼭지점(seed vertex)을 이용하여 꼭지점 연결성(connectivity) 정보를 재배열하는 전처리를 수행한다. 제안된 방법은 워터마크 검출 과정에서 원본 메쉬를 사용하지 않는 블라인드 검출(blind detection)이 가능하다.

제안된 두번째 방법은 통계적 특성에 기반한 방법이다. 웨이블릿 변환 후, 스케일 계수(scale coefficients) 벡터의 크기값 분포를 균일한 구간 (bin)으로 나눈다. 각 구간에 속하는 스케일 계수 벡터 크기값의 분포를 변화시키기 위해서, 제안된 방법은 삽입하고자 하는 워터마크 신호에 따라 히스토그램 대응 함수를 이용하여 각 구간의 평균값(mean value) 또는 분산(variance)을 수정한다. 제안된 워터마크 검출 방법은 워터마크 삽입과정과 동일하게 웨이블릿 변환 후 스케일 계수의 분포를 이용하여 검출할 수 있을 뿐만 아니라, 삽입된 워터마크가 공간 영역에서 확산되어 존재하기 때문에 웨이블릿 변환 과정 없이 공간영역에서 꼭지점 좌표 크기값(vertex norm)의 분포를 이용하여 검출하는 것이 가능하다. 제안된 두번째 방법은 첫번째 방법보다 위상 공격(topological attacks)에 더욱 강인함을 보였다.

3차원 동형 메쉬 시퀀스 워터마킹

본 논문에서는 시간축 웨이블릿 변환을 이용한 3차원 동형 메쉬 시퀀스 워터마킹 기법을 제안한다. 제안된 첫번째 방법은 꼭지점 좌표값의 세 축(axis)에 독립적으로 워터마크를 삽입하고, 두번째 방법은 꼭지점 좌표를 구면 좌표계로 변환 후 좌표 벡터 크기값에만 워

Acknowledgments

This work is done by cooperative Ph. D program between Institut National des Sciences Appliquées de Lyon (INSA Lyon), France and Yeungnam University, Korea.

Contents

Chapter 1

General Introduction

1.1 Introduction

The recent growth of digital technologies has enabled the perfect reproduction, ease of editing, and the Internet distribution of multimedia data. Consequently, they also have brought concerns of copyright infringement, illegal distribution, and unauthorized tampering. Multimedia information hiding techniques started to come out to relieve these concerns. Traditional data protection techniques such as encryption are not adequate for copyright enforcement, because the protection cannot be ensured after the data is decrypted by authorized users. Unlike the encryption, digital watermarking does not restrict access to the host data, but ensures the hidden data to remain inviolated and recoverable [COX02, FRID02, PODI01]. Most of previous watermarking technologies have focused on traditional media data, such as digital audio, image, and video data. Recently, 3-D polygonal models have been widely used in many fields which need realistic visualization of the object, for example, CAD (Computer Aided Design), character animation in movie, video games, medical objects and so on. Despite such popularity, few watermarking methods have been proposed for 3-D geometric model. This is caused by, in part, the watermarking technology that has emerged for image, video, and audio which cannot be easily adapted to work for 3-D geometric models. This thesis deals with digital watermarking methods for 3-D surface meshes and mesh sequences for the application of copy-right protection.

1.2 Organization of the Thesis

Before presenting our proposed methods, we address the background of 3-D surface meshes and mesh sequence in Chapter 2. A state of the art of digital watermarking of 3-D meshes and mesh sequences is presented in Chapter 3. In particular, Chapter 3 describes our previous watermarking methods using the distribution of vertex norms [CHO04, CHO07] and previous wavelet transform based watermarking methods [KANA98, UCCH04] which have inspired our proposals.

Our proposed watermarking methods for 3-D meshes and mesh sequences are introduced in Part I and Part II, respectively: Digital Watermarking for 3-D Surface Meshes (Part I), and Digital Watermarking for 3-D Mesh Sequences (Part II).

In Part I, we propose wavelet analysis based digital watermarking methods for 3-D triangular surface meshes. Previous wavelet analysis based methods [KANA98, UCCH04] can process only regular meshes. Our proposals can be applied to irregular as well as regular meshes by using recently introduced irregular wavelet analysis scheme [VALE04a].

In Chapter 4, L2 norm of the wavelet coefficients (high frequency component) at multiresolution levels is modified by spread spectrum method [COX97, HART98]. The proposed method is designed for blind watermark detection, which can extract the watermark without reference of cover mesh model. To ensure the connectivity synchronization after inverse wavelet transform, we introduced a vertex and face re-ordering process as pre-processing in both watermark embedding and extraction. The re-ordering process is also able to be used to extract the error-free watermark after random connectivity reordering attacks. Since L2 norm of the wavelet coefficients is invariant to translation and rotations, the method is also robust against rotation and translation.

In Chapter 5, we introduce a blind watermarking method which is robust against topological attacks. When the watermark information is embedded into wavelet coefficients arranged in a certain order, it is difficult to extract the watermark without the original meshes after the connectivity information is destroyed by topological attacks. In this proposal, statistical features of scale coefficients on an approximation mesh (low frequency) are used for watermark embedding. The histogram of L2 norm of scale coefficients is divided into distinct bins of equal range. Note that, each bin is used as unit to embed one bit of the watermark in our proposal. The distribution of each bin can be approximated to uniform distribution. Then, the distribution of each bin is modified according to the watermark bits to be embedded. The mean or the variance of the bin is modified according to the watermark by histogram mapping function [CHO07]. These techniques allow detecting the watermark without referring to the original meshes. The hidden watermark is extracted, not only from the same resolution level as used in embedding process, but also from the spatial domain. Our proposal does not require the original meshes, in addition, any pre-processing such as registration and re-sampling is not needed.

In Part II, we present digital watermarking methods for 3-D mesh sequences. The main idea of the proposals is to modify the statistical features of the distribution of temporal wavelet coefficients. Two proposals are introduced in Chapter 6.

The first proposal embeds the watermark into the temporal wavelet (high frequency) coefficients of each coordinates of the sequence. The high frequency coefficients from temporal wavelet transform can be approximated to Laplacian probability density function with symmetric property. High frequency coefficients is changed to have asymmetric distribution by histogram mapping function according to the watermark bits to be embedded. Two methods are introduced. The watermark is embedded into each axis of the high frequency coefficients. The distribution of temporal scale (low frequency) coefficients which are less sensitive to general attacks is utilized in the bin generation procedure. In the second method, the Cartesian coordinates of the vertices of the input mesh sequence are converted to the spherical coordinates and L2 norm of each vertex is wavelet transformed along the time axis. Then the watermark is embedded by modifying the distribution of the wavelet coefficients with the same method used in the first method. After inverse wavelet transform of the modified L2 norms, the spherical coordinates are converted to Cartesian coordinates. By using the distribution, our method can retrieve the hidden watermark without any information about original mesh sequences in the process of watermark detection.

The second proposal embeds the watermark into low frequency coefficients as well as high frequency coefficients. After temporal wavelet transform, the watermark is embedded into each low frequency frame by using the variance modification method used in [CHO07], and the watermark is also embedded into temporally high frequency coefficients by using the first proposal. This method allows extract the watermark from each low frequency frame and the distribution of high frequency coefficients. In addition, it is possible to extract the watermark from a single frame in spatial domain.

We conclude the thesis with final remarks and suggestions for further studies in Chapter 7.

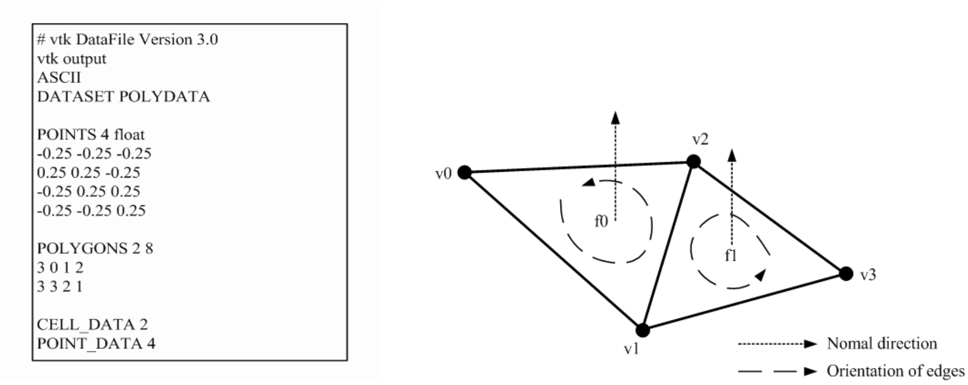
Chapter 2

Background

In this chapter, the background on the researches of this thesis are represented. We introduce the notations and data structure of 3-D surface meshes. We explain regularity of vertex and meshes. We describe several operations for 3-D surface meshes which used to evaluate the performance of our proposals. Regular and irregular subdivision scheme for wavelet analysis of 3-D triangular meshes are also explained. Note that the irregular wavelet analysis is utilized to embed the watermark into 3-D static meshes in Part I of this thesis. The history of digital watermarking and its applications and requirements are introduced. The watermarking methods are classified into spatial domain and frequency domain.

2.1 3-D Surface Meshes and Mesh sequences

Digital models of 3-D objects have been represented in the form of triangular surface meshes. In various representation tools, triangular meshes are usually represented by two



```
# vtk DataFile Version 3.0
vtk output
ASCII
DATASET POLYDATA

POINTS 4 float
-0.25 -0.25 -0.25
0.25 0.25 -0.25
-0.25 0.25 0.25
-0.25 -0.25 0.25

POLYGONS 2 8
3 0 1 2
3 3 2 1

CELL_DATA 2
POINT_DATA 4
```

(a) Example of the vtk file format

(b) Visualization of Fig. 2.1(a)

Figure 2.1: General representation of 3-D triangular surface meshes

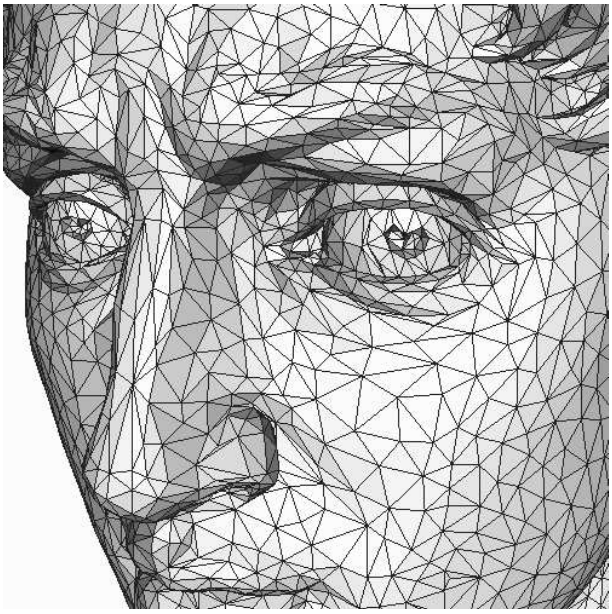
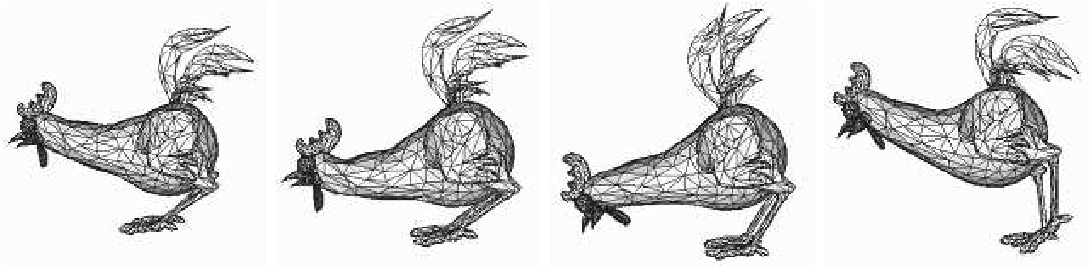


Figure 2.2: Zoom of a triangular surface meshes from the face of "Davidhead"

components, namely the geometry and the connectivity. Fig. 2.1 shows general representation of 3-D object by triangular surface meshes, where vtk (*Visualization Tool Kit* [SCHR02]) file format is given. The geometry information is a set of floating point values of vertex coordinate which is represented in Cartesian coordinate system. Connectivity information defines a set of triangles from geometry information to represent the surface of the object. Connectivity information also tells the normal direction of the face and the orientation of the edges. Fig. 2.2 shows and example of a triangular surface meshes, "Davidhead".



(a) 1st frame (b) 30th frame (c) 60th frame (d) 90th frame

Figure 2.3: "Chicken" mesh sequence

A mesh sequence is also called dynamic mesh while a 3-D surface meshes is called static mesh. As a video data consists of successive still images, a mesh sequence consists of successive static mesh models. Mesh sequences are classified by topological change between the adjacent static mesh frames: isomorphic and non-isomorphic (also called hetero-morphic) mesh sequences. In the case of isomorphic mesh sequence which is used in this thesis, each static mesh frame has the same connectivity and only variation of geometrical position of the vertices. Fig. 2.3 shows an example of isomorphic mesh sequence, "Chicken".

2.1.1 Definitions and Notations: Triangular Meshes

A triangle mesh is represented by two components. There are many kinds of notations to describe the connectivity and geometry of meshes [SPAN66, FREY99, HORM02]. The definitions and notations for triangular mesh follow Hormann [HORM02] in this thesis. The geometric information is given as a set of vertices with their 3D coordinates. The topological (or connectivity) information explains how the vertices are connected.

Let $V = \{v_0, v_1, \cdots, v_{|V|-1}\}$ be a set of $|V|$ vertices where,

$$v_i \;=\; [x_i \;\; y_i \;\; z_i\;]^t \tag{2.1}$$

and a set of $|T|$ triangles,

$$T = \{t_0, t_1, \cdots, t_{|T|-1}\}, \tag{2.2}$$

where each triangle is represented as a triplet of vertex indices, $t_i = (t_0^i, t_1^i, t_2^i)$ with $t_k^i \in V$. Here, $|V|$ and $|T|$ indicates the cardinality of vertices and triangles, respectively. Then, a triangular surface meshes M consists of a pair of (V, T).

The edges set E in the mesh is defined by

$$E = E_I \cup E_B, \tag{2.3}$$

where E_I , E_B is the sets of interior and boundary edges, respectively. In the case of manifold meshes, an interior edge is shared by two adjacent triangles, while boundary edge does not have adjacent triangle.

The boundary B is defined by a set of boundary edges E_B. When the union of all boundary edges is a set of closed simple boundary polygons, we denote the number of boundary $|B| \geq 0$. For a triangular meshes, the genus G of triangular meshes is defined by the number of handles of a mesh M. If M is topologically equivalent to a sphere or a disk, then $G = 0$, if M is topologically equivalent to a torus, then $G = 1$.

The *Euler formula* for general polyhedra is

$$|V| - |E| + |F| = 2 - 2G - |B|\,, \tag{2.4}$$

where $|V|\,, |E|\,, |F|$, and $|B|$ are the numbers of vertices, edges, faces, and boundary polygons and G is the genus of the polyhedron. For triangle meshes ($|F| = |T|$) we observe

$$3|T| = 2|E_I| + |E_B| \tag{2.5}$$

by counting the edges of the triangles. Since every boundary edge is adjacent to two boundary vertices and every boundary vertex has exactly two adjacent boundary edges we further have

$$|E_B| = |V_B| \tag{2.6}$$

and rewrite Eq. 2.4 as

$$|V| - \frac{1}{2}(|V_B| + |T|) = 2 - 2G - |B|\,. \tag{2.7}$$

2.1.2 Neighbor vertex and neighbor triangle

If two vertices v and w are connected by one edge, v is a neighbor vertex of w and vice versa. Similarly, if two triangles, t and s, share a common edge, t is a neighbor triangle of s and vice versa. Fig. 2.4 depicts the neighbor vertices in n-ring structure. For the seed vertex indicated by black small circle v, the neighbor vertices of the seed vertex v is called 1-ring vertices. Here, 1-ring vertices and their neighbor vertices are called 2-ring neighbor vertices.

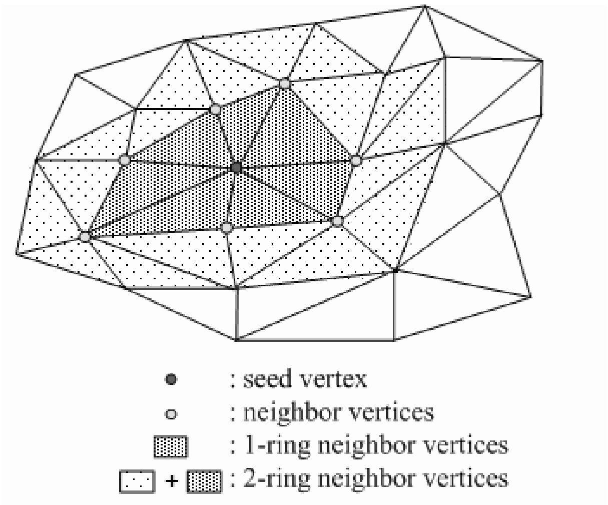

● : seed vertex
○ : neighbor vertices
▩ : 1-ring neighbor vertices
⬚ + ▩ : 2-ring neighbor vertices

Figure 2.4: 1-ring and 2-ring neighbor vertices

2.1.3 Valence of a vertex, regular/irregular meshes

For a vertex v, the number of v's 1-ring neighbor vertices is called the valence (or degree) of v and denoted $d(v)$. Note that $d(v)$ equals the number of adjacent triangles if v

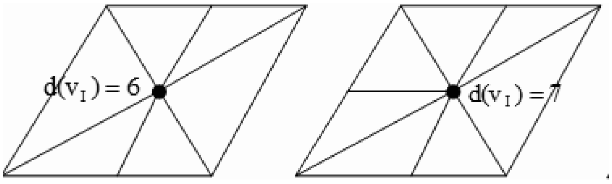
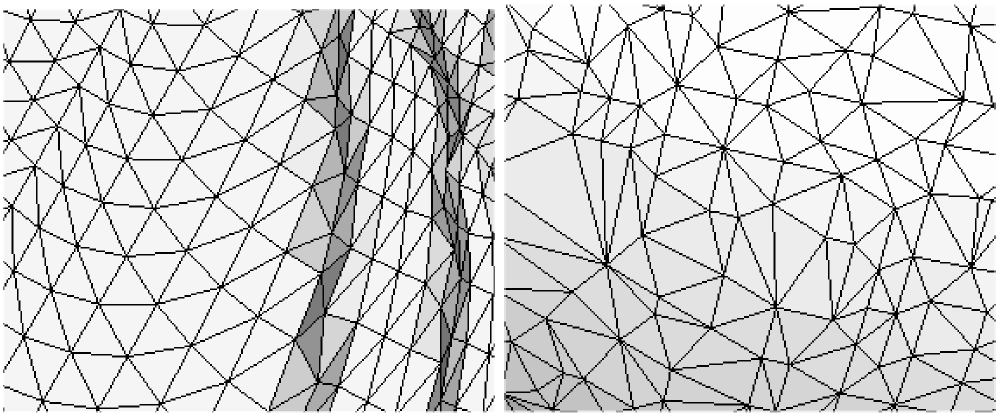


Figure 2.5: Valence of a vertex , regular vertex on the left and irregular vertex on the right.



(a) Regular meshes

(b) Irregular meshes

Figure 2.6: Regularity of the meshes

is an interior vertex.

Regularity of the vertex v is defined by the valence of the vertex. v is regular, if $d(v) = 6$ for $v \in V_I$ or $d(v) = 4$ for $v \in V_(B)$. Otherwise, v is irregular. Fig. 2.6 shows an examples of valence of a vertex.

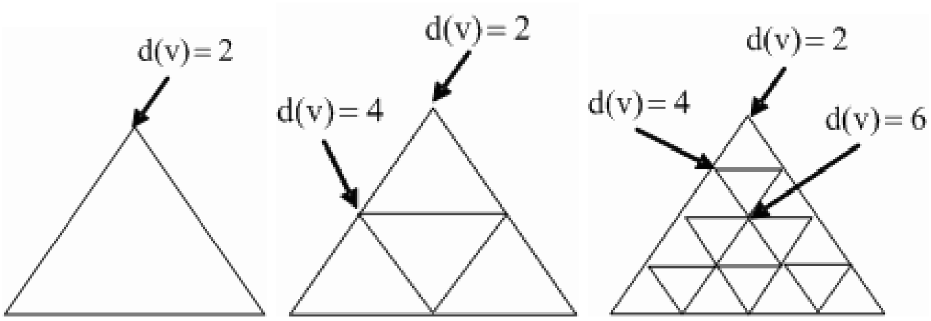
Similarly, a meshes is regular (Fig. 2.6-(a)) if the meshes consists of regular vertices. Otherwise, irregular meshes (Fig. 2.6-(b)). Note that a meshes is semi-regular when the most of vertices of the meshes have valence 6 [LEE05].

2.1.4 Regular Subdivision of Meshes

1-to-4 subdivision of meshes is called regular subdivision. Fig. 2.7 illustrates the procedures of 1-to-4 regular subdivision [LOOP87, LOUN94]. The initial mesh in Fig. 2.7-(a), is also called base mesh. Triangles of the base mesh is divided into four triangles in Fig. 2.7-(b),(c). The valences of vertices never changes by the 1-to-4 subdivision. $d(v)$ is 6 for interior vertices and 4 for vertices of boundaries. Fig. 2.8 is an example of 1-to-4 subdivision of 3-D surface meshes. The valences of interior and of boundary vertices are also kept in the case of 3D surface meshes. Note that re-meshing is required to process the irregular meshes [LOUN94, VALE04a].

2.1.5 Irregular Subdivision of Meshes

Valette and Prost [VALE04a] are introduced the irregular subdivision scheme to subdivide irregular meshes as well as regular and semi-regular meshes. Fig. 2.9 shows the example of the irregular subdivision (1-to-4, 1-to-3, 1-to-2 and 1-to-1). Total of 11 cases of subdivision are possible. The irregular subdivision scheme is used for simplification and reconstruction of the connectivity of the input meshes during the wavelet analysis. Note that our watermarking methods introduced in Part I utilize the irregular subdivision sch-



(a) Basemesh, M^0 (b) M^1 (c) M^2

Figure 2.7: 1-to4 regular subdivision of meshes [LOOP87, LOUN94]

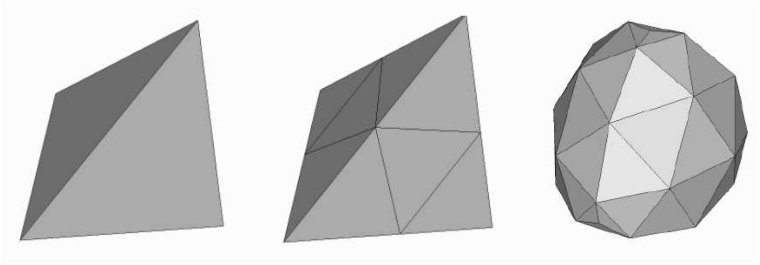


Figure 2.8: 1-to-4 regular subdivision of 3D surface mesh. Left: basemesh (tetrahedron), middle: 1-to-4 subdivision mesh, and right: geometrical deformation by Loop [LOOP87] filter

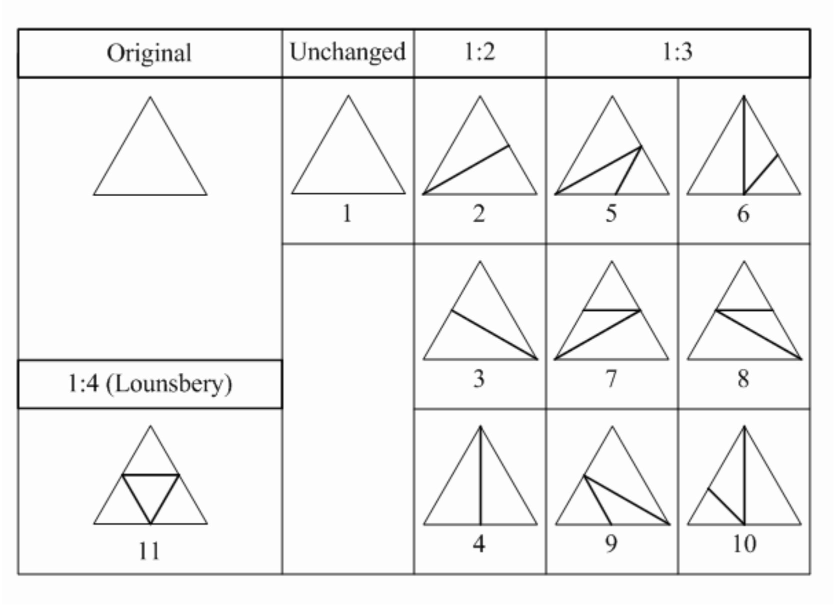


Figure 2.9: Irregular subdivision in [VALE04a]

eme to embed the watermark into both semi-regular and irregular meshes without using re-meshing.

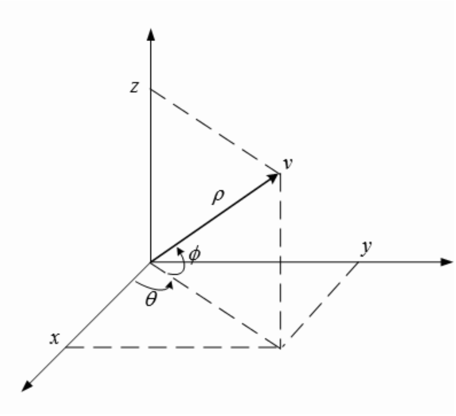


Figure 2.10: Coordinate system (x, y, z) and (ρ, θ, ϕ)

2.1.6 Conversion of Coordinate System

Each vertex v_i from a set of vertices V of the surface meshes, is represented by its Cartesian coordinates $v_i = (x_i, y_i, z_i)$ and it can be converted into spherical coordinates $v_i = (\rho_i, \theta_i, \phi_i)$ by

$$\begin{aligned} \rho_i &= \sqrt{(x_i)^2 + (y_i)^2 + (z_i)^2} \\ \theta_i &= tan^{-1}\frac{y_i}{x_i} \\ \phi_i &= cos^{-1}\frac{z_i}{\sqrt{(x_i)^2 + (y_i)^2 + (z_i)^2}} \qquad , \end{aligned} \tag{2.8}$$

where ρ_i is the distance from the origin to the vertex v_i and azimuth θ_i and elevation ϕ_i are angular displacements in radians measured from the positive x-axis, and the x-y plane, respectively. (Fig. 2.10).

Spherical coordinates can be converted into Cartesian coordinates by

$$\begin{aligned} x_i &= \rho_i \cos\theta_i \sin\phi_i \\ y_i &= \rho_i \sin\theta_i \sin\phi_i \\ z_i &= \rho_i \cos\phi_i. \end{aligned} \tag{2.9}$$

2.1.7 Quality measure of 3-D surface meshes

Distance between two surface meshes can be easily measured by MSE (Mean Square Error) or SNR (Signal-to-Noise Ratio) which are widely used for quality measurement of 2D images or audio signal. However, if the two meshes have different topology or different numbers of the vertices, we have to consider another quality measurement technique. In this subsection, we introduce two distortion measures that have been often used in 3-D mesh processing: Hausdorff distance [CIGN98], root mean square distance.

Hausdorff distance

Hausdorff distance is defined as a vertex-to-surface distance. Given a vertex v and a set of vertices V in a surface meshes, the distance is defined as,

$$e(v, V) = \min_{v' \in V} d(v, v'), \tag{2.10}$$

where $d(v, v')$ is the euclidean distance between two points. The one-side distance between two surfaces V and V' is then defined as,

$$E(V, V') = \max_{v \in V} e(v, V'). \tag{2.11}$$

Note that this definition of distance is not symmetric. There exist surface such that $E(V, V') \neq E(V', V)$ [CIGN98, ASPE02]. A two-sided distance, so called Hausdorff distance [CIGN98] is defined by taking the maximum of $E(V, V')$ and $E(V', V)$.

$$E_H(V, V') = \max\{E(V, V'), E(V', V)\}. \tag{2.12}$$

Root Mean Square distance

Given a set of uniformly [ASPE02] or randomly [CIGN98] sampled mesh faces, we define the mean distance E_{mean} between two surfaces as the surface integral of the distance divided by the area of V:

$$E_{mean}(V, V') = \frac{1}{Area(V)} \int \int_V e(v, V') dV_{v \in V} \tag{2.13}$$

From Eq. 2.10 and Eq. 2.13, a root mean square distance is defined as,

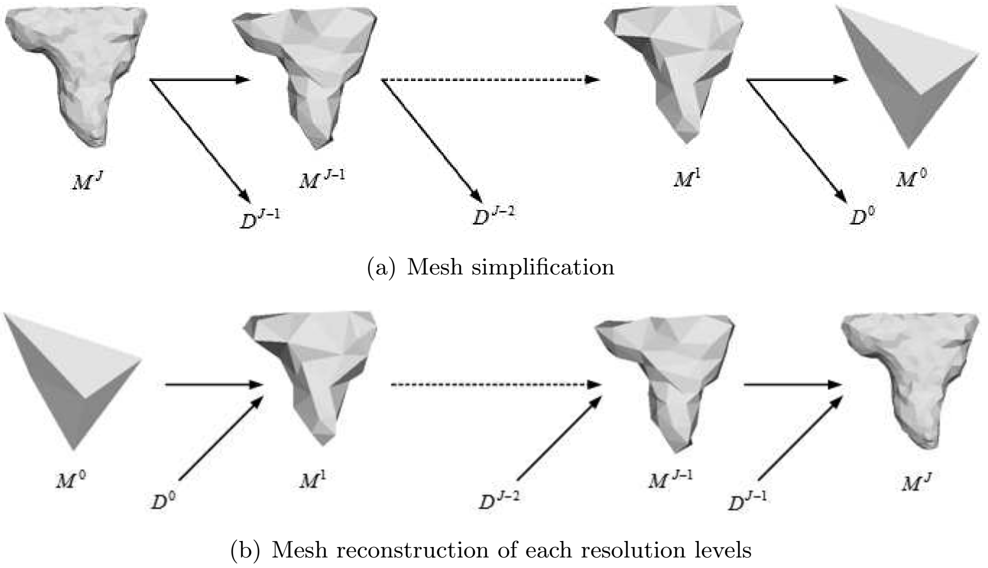
$$E_{rms}(V, V') = \sqrt{\frac{1}{Area(V)} \int_V e(v, V')^2 dV_{v \in V}} \tag{2.14}$$

Note that we utilize the maximum root mean square distance for the quality measurement of our watermarking methods throughout this thesis.

$$E_{mrms}(V, V') = \max\{E_{rms}(V, V'), E_{rms}(V', V)\} \tag{2.15}$$

2.2 Wavelet Analysis for 3-D Surface Meshes

Multiresolution analysis of 3-D object is receiving a lot of attention due to the practical interest of 3-D modeling in a wider and wider range of applications, such as Computer Graphics and Computer-Aided Design. Multiresolution analysis of these objects gives some useful features: Several levels of detail can be built for these objects, accelerating the rendering when there is no need for sharp detail, and allowing progressive transmission. Wavelet analysis, one of the most useful multiresolution representation techniques, have been used in a broad range of applications, including image compression, physical simulation, hierarchical optimization, and numerical analysis. The basic idea behind wavelet analysis is to decompose a complicated function into a simpler coarse-resolution part, together with a collection of perturbations called wavelet coefficients. Lounsbery [LOUN94] extended wavelet analysis to be applied to 3D surface meshes. In this subsection, we describe briefly the wavelet analysis and some matters to be investigated for blind watermarking.



(a) Mesh simplification

(b) Mesh reconstruction of each resolution levels

Figure 2.11: Multiresolution analysis of 3-D surface meshes

The wavelet multiresolution analysis of 3-D model has been introduced in [LOUN94] for semi-regular meshes. Given a coarse triangulation, denoted the base mesh M^0, a semi-regular mesh is constructed by performing repeated quadrisection on all triangles (one-to-four subdivision). The coarse mesh vertices have arbitrary degree (valence) while the subdivided mesh interiors and boundaries vertices have degree six and four, respectively (See subsection 2.1.6). A genus zero mesh has almost tree irregular vertices (tetrahedron), a genus one mesh e. g. homeomorphic to a torus, or a mesh with two boundaries, e. g. homeomorphic to a cylinder, can be completely regular.

The multiresolution representation is two complementary nested sets of vector spaces:

the approximation spaces,

$$V^0 \subset V^1 \subset \ldots \subset V^{J-1} \subset V^J, \tag{2.16}$$

and the detail spaces,

$$W^0 \subset W^1 \subset \ldots \subset W^{J-1}, \tag{2.17}$$

where $V^j = V^{j-1} \oplus W^{j-1}$. We denote by zero the coarsest space index and finer spaces have higher index.

The multiresolution mesh set is a hierarchy of approximation

$$M^0 \in V^0, M^1 \in V^1 \ldots V^J \in M^J, \tag{2.18}$$

where the highest level is that of the original mesh. Note that the highest level has no approximation space. For the approximation vector space and the detail vector space the basis function are the scaling function and the wavelets, respectively. As V^{j-1} and W^{j-1} are both subspaces of V^j the scaling function in V^{j-1} and the wavelets in W^{j-1} are linear combination of the scaling function in V^j denoted by matrix P^j and Q^j, respectively. In addition $V^j = V^{j-1} \oplus W^{j-1}$, it follows that the approximation coefficients of the geometry (scaling coefficients) C^j ($C = \begin{bmatrix} v_0 & v_0 & \cdot & v_{|v|-1} \end{bmatrix}^t$) at the resolution j can be calculated from the coarser approximation coefficient C^{j-1} and detail coefficients (wavelet coefficients) D^{j-1} by the synthesis formula,

$$C^j = P^j C^{j-1} + Q^j D^{j-1}. \tag{2.19}$$

For the analysis, e. g. for the calculation of the approximation and detail coefficients of M^j at resolution $j-1$, we remain that $V^{j-1} \in V^j$ and $W^{j-1} \in V^j$ the scaling function in V^{j-1} and the wavelets in W^{j-1} are linear combination of the scaling function in V^j denoted by matrices A^j and B^j, respectively.

$$C^{j-1} = A^j C^j \tag{2.20}$$

$$D^{j-1} = B^j C^j \tag{2.21}$$

The matrices A^j, B^j, P^j and Q^j, define the analysis and reconstruction filter-bank, respectively. Note that sub-sampling is incorporated in the matrix operations. To ensure the exact reconstruction of M^j from M^{j-1} and D^{j-1} , the filter-bank must satisfy:

$$\left[\frac{A^j}{B^j} \right] = \left[P^j \mid Q^j \right]^{-1} \tag{2.22}$$

First, we only consider the Lounsbery scheme where the mesh as a 1:4 subdivision connectivity e.g. the mesh hierarchy can be considered as successive quadrisections of the base mesh (M^0) faces followed by deformation of edges midpoints to fit the surface to be approximated. Conversely, four-to-one face coarsening is the inverse operation of quadrisection. In this scheme the wavelets functions are hat functions associated with odd vertices of the mesh at resolution j and linearly vanishing on the opposite edges. This wavelet is often called the "Lazy wavelet". The scaling function are also hat function but with a twice wider support and associated with the even vertices. Note that at the current resolution j the Lazy wavelets are scaling functions at resolution $j+1$.

Clearly the wavelet function basis is orthogonal but not that of the scaling function. In addition wavelets are not orthogonal to the scaling functions. It results a poor approximation of the surface modeled by M^j at level $j-1$. The mesh M^{j-1} is just the sub-sampled mesh M^j. Note that theses wavelets are used in the watermarking scheme [KANA98] and [UCCH04].

The Lazy wavelet can be improved by making it more orthogonal to the scaling function by means of the Lifting scheme [SWEL95]. This is done by subtracting off some amount of the coarser scaling functions from each wavelet. This 'amount' is introduced by coefficients α_i^j in the k-disc of the vertex of interest. The last step in calculated the new wavelet is to make it orthogonal to scaling function on the k-disc by solving an equation set in the least-square sense for the unknown coefficients α_i^j. It results a new (index lifting) filter bank as follows:

$$A^j = A^j_{lazy} + \alpha^j . B^j_{lazy} \tag{2.23}$$

$$B^j = B^j_{lazy} \tag{2.24}$$

$$P^j = P^j_{lazy} \tag{2.25}$$

$$Q^j = Q^j_{lazy} - P^j_{lazy} . \alpha^j \tag{2.26}$$

Recently the wavelet multiresolution analysis has been extended to irregular mesh (vertices can have any degree) by Valette and Prost [VALE04a]. This new approach allows direct processing of the output mesh of isosurface extracted by usual algorithms such as the well-known Marching Cubes [LORE87]. Due to the irregular connectivity of the connectivity graph G the surface analysis by wavelets cannot be implemented without the knowledge of face fusions needed for mesh coarsening. Coarsening is the inverse problem of successive subdivision of the base mesh. In sharp contrast with Lounsbery regular 4 to 1 faces merge, in the irregular approach faces where merged 4:1, 3:1, 2:1, or 1:1 (no merge at the current level). Fig. 2.9 shows an example of irregular subdivision. In addition edge flips where used to improve the merging efficiency. In Wavemesh [VALE04a] face merge coarsening results in the coding of the connectivity graph G with side information of edge flips. The merging algorithm was based on region growing starting from a seed group of four faces merged into one faces. In the particular cases of semi-regular mesh the seed group is chosen with an irregular vertex on one of its corner in order to inverse the subdivision of the base mesh [VALE03]. After this first step the filter bank (Eq. 2.23-2.24) can be constructed and both the approximation and detail coefficients can be computed from the highest level J to the lowest level 0. Such stage takes into account of the merging factor and the connectivity graph for both the matrix entries construction and the scalar product definitions used in the lifting scheme. Similarly the reconstruction (synthesis) from level 0 to J results from the construction of the filter bank (Eq. 2.25-2.26) and filtering [VALE04a, VALE04b] at each level. Note that the irregular wavelet analysis is applied to watermarking methods for 3-D surface meshes in Part I of this thesis.

2.3 Digital Watermarking

In this section, we explain the historical background of digital watermarking. The applications (copyright protection, ownership identification, certification, authentication, conditional access, copy-control, captioning and broadcast monitoring) and requirements of digital watermarking are addressed. We classified the watermarking methods according to the type of watermark (visible/invisible), the embedding domain(spatial/frequency domain) and the detection method (blind/non-blind). We give a brief overview for watermark embedding and extraction algorithms.

2.3.1 History

The idea of information hiding can be traced back to a few thousand years ago. As surveyed in previous papers [PETI99, COX02], simply encrypted the content of messages is not always suitable in practice. In many rivalry environments, concealing the existence of communication is desirable to avoid suspicion from adversaries. Watermarking stemmed from the study of Steganography. The word "Steganography" originated from old Greek language is still in use today. Steganography can be translated as "cover writing". It was basically a way of transmitting secret messages between allies and was used as early as 1000 BCE. First references to steganography appear in Homer's "Iliad" and "Histories of Herodotus" (440 BCE) [KEJA03, HERO03]. Demeratus sent a warning about a forthcoming attack to Greece by writing it on a wooden panel and covering it in wax. Wax tablets were in common use then as re-usable writing surface, sometimes used for shorthand. Another ancient example is that of Histiaeus [HIST], who shaved the head of his most trusted slave and tattooed a message on it. After his hair had grown the message was hidden. The purpose was to test a rebellion against the Persians. Digital watermarking takes its name from watermarking of paper or money as a security measure. Watermarks first introduced in Bologna, Italy in 1282. They have been used by paper-makers to identify their product, and also on postage stamps, currency, and other government documents to discourage counterfeiting. Digital watermarking is not a form of steganography, in which data is hidden in the message without the end user's knowledge, although some watermarking techniques have the steganographic feature of not being perceivable by the human eye. While the addition of the hidden message to the signal does not restrict that signal's use, it provides a mechanism to track the signal to the original owner. Tirkel *et al.* introduced and defined the word "digital watermarking" and its related terminologies in [VANS94].

2.3.2 Application and Requirements

The first application of digital watermarking is related to copyright protection of digital media. Digital watermark is designed to permanently reside in the host data without appending the header information, then the description about data can be embedded into the data itself instead of using the metadata [1]. Based on the idea of metadata, the watermarking technologies are available in ownership identification, for example, a real system is implemented for International Science Information (ISI)'s Electronic Library Project [MINT97].

In the applications of data security, watermarks may be used for certification, authentication, conditional access and copy-control. Digital watermarking allows to mutually link information on the documents. For example, the name of a passport owner is normally printed in clear text and is hidden as an invisible watermark in the photo of the owner. If anyone would intend to forge the passport by replacing the photo, it is possible to notice the forgery be scanning and detecting the watermark in the photo. Verification of authenticity of digital media may be done by fragile watermarking [FRID01, CAYR03]. The objective of verification includes the inspection of tampered area of the content. Conditional access or copy-control to confidential data on CD/DVD-ROMs is possible by using the digital watermark on the CD/DVD label which contains the key to decrypt data in CD/DVD [COX02]. The main idea is that someone has to scan and detect the watermark on the label before accessing to confidential data in CD/DVD. If someone copy CD/DVD, he will not be able to read the clear-text in CD/DVD since he does not have the required watermark.

'The invisible information within data itself' is very attractive characteristic of the digital watermark in various application as well as copyright protection. The captioning of the digital media together with associated information could be easily achievable with the invisible watermark. A good example for captioning is songs played on the radio. When the music owners and the radio stations, are interested in an accurate count of exactly what gets played on the air. If the radio station plays a song with inaudibly watermarked information, it enables an automatic radio listener monitoring for both radio station and the music owner. In addition, this is also an interesting application for the sponsors of television and radio stations. The stations can use the watermark of their sponsors and the watermark will be embedded into the advertisement. Both the stations and the sponsors can detect the watermark to log automatically that have been played. This application is also called broadcast monitoring.

[1] Structured, encoded data that describe characteristics of information-bearing entities to aid in the identification, discovery, assessment, and management of the described entities.

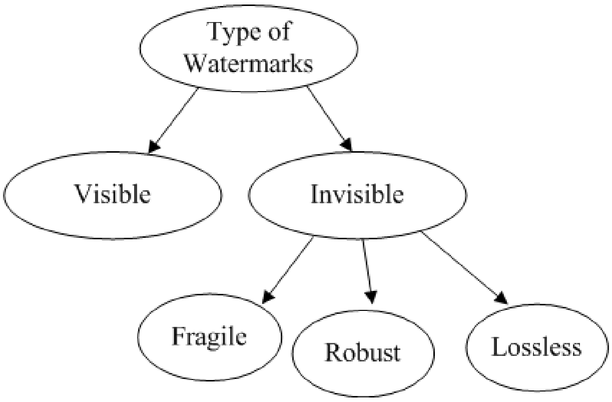


Figure 2.12: Type of watermarks

There are three common requirements for the most watermarking applications: robustness, invisibility and capacity. *robustness* describes how well watermarks survive on intentional (e.g. cropping a part of the image) or non-intentional (e.g. lossy compression) signal processing operations, *invisibility* describes how imperceptible the watermarks are. *capacity* that is the number of watermark bits to be embedded is also an important property of the digital watermarking. The relative importance of these properties depends on the application of the given watermarking system. For example, in applications where we have to detect the watermark in a copy of a content that has been broadcast over an analog channel, the watermark must be robust against the degradation caused by that channel. However, if we can reasonably expect that a content will not be modified at all between embedding and detection, the watermark's robustness is irrelevant [COX02].

2.3.3 Classification of Watermarking Methods

There are many ways to categorize digital watermarking techniques. In a straightforward way, it can be classified into visible and invisible according to the type of the watermarks (Fig. 2.12). Visible watermarks change the signal altogether such that the watermarked signal is totally different from the actual signal, e.g., adding an image as a watermark to another image. Stock photography agencies often add a watermark in the shape of a copyright symbol (©) to previews of their images, so that the previews do not substitute for high-quality copies of the product included with a license. Invisible watermarks do not change the signal to a perceptually great extent, i.e., there are only minor variations in the watermarked signal. In contrast to robust watermarks which are designed to be remained under the severe changes to the image, fragile watermarks are made to be easily broken under the slight changes. Semi-fragile watermarks are designed to break under all changes that exceed a user-specified threshold. Another form of watermarking, called lossless (or reversible) watermarking, have been introduced [FRID01, FRID02], and is designed to recover the original unmodified signal after detecting watermarks.

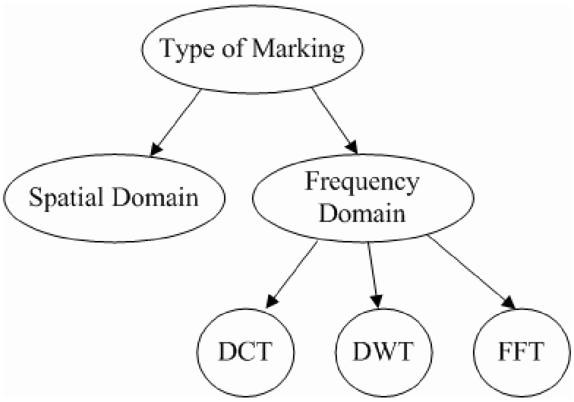


Figure 2.13: Type of watermarking

The watermarking techniques also can be classified into spatial domain methods and frequency domain methods as shown in Fig. 2.13. Most of early watermarking methods were developed in the spatial domain. Spatial domain watermarking methods embed the watermark by directly modifying the host signal (e.g. an amplitude of pixel for 2-D image). In 1993, Tirkel *et al.* [TIRK93] proposed a digital watermarking method for image data using the least significant bit (LSB) plane modification. Fig. 2.14 shows an example of LSB modification. The image data should compress each pixel into 7-bits using adaptive histogram manipulation to get the most effective gain. But, the LSB watermarking is not robust against most signal processing even with small change. After this, several spatial domain techniques [LANG97, PITA98, QUIS95] have been introduced, but they are still vulnerable to common signal processing and simple cropping operation. This is the reason why the spatial domain methods have not widely used in copyright protection applications. Spatial domain methods have been often used as fragile watermarking for authentication.

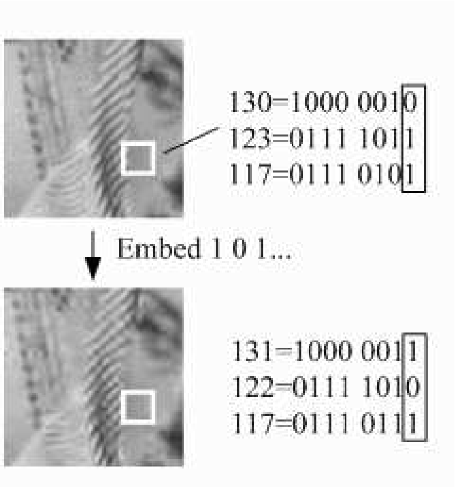


Figure 2.14: An example of LSB modification in [TIRK93]

Frequency domain watermarking methods decompose the host signal into transformed domain by using discrete cosine transform (DCT), discrete wavelet transform (DWT) or discrete fourier transform (DFT), and embed the watermark by modifying the transformed coefficients. Frequency domain watermarking is useful for taking advantage of perceptual criteria in the embedding process, for designing watermarking techniques which are robust to common compression techniques, and for direct watermark embedding of compressed bit streams. A common transform framework for image is the block-based DCT which is fundamental building block of current image coding standards such as JPEG (Joint Photographic Expert Group) and video coding standards such as MPEG (Motion Picture Expert Group) video coder and the ITU (International Telecommunication Union) H.26x codecs. Koch *et al.* [KOCH96] introduced a DCT based image watermarking method. To be robust against common signal processing attacks, the method modifies the middle frequency coefficients of 8×8 block DCT coefficients. They show that the method is robust against JPEG quality factor as low as about 50%. Cox *et al.* [COX97] introduced spread spectrum approach borrowed from communication theory. The authors consider the watermarking process in digital image as a spread spectrum communication in noisy channel. They show that the method is very effective both in terms of image quality and the robustness to carious intentional and non-intentional signal processing attacks. One of the most significant contributions in this work is the realization that the watermark should be inserted in the perceptually significant portion of the image in order for it to be robust to such attacks. After DCT, the watermark that has normal distribution with zero mean and unit variance was spread over 1000 visually most significant coefficients except DC component. The strength factor of watermark is scaled according to the strength of the host-signal of particular frequency components. The authors suggested to set the strength factor to 0.1 to provide a good tradeoff between imperceptibility and robustness. Seo *et al.* [SEO01] proposed a DWT based watermarking method, which can be conveniently integrated in JPEG-2000 baseline. Usual DCT/DWT based watermarking techniques [COX97, PODI98] insert a watermark into the coefficients after completed the transform, while the method a watermark into the coefficients obtained from ongoing process of lifting for DWT. The method allows to selectively determine frequency characteristic of the coefficients where watermark is embedded, so that the watermark cannot be easily removed or altered even when filter-bank for DWT is known. Note that our proposal introduced in Chapter 4 embeds the watermark into wavelet coefficients by using spread spectrum approach [COX97].

The watermarking system can be classified into informed detection (also called non-blind detection) and blind detection according to detection procedure. Although it has been known that blind schemes are less robust than informed ones, they are more useful for various applications where a host signal is not available in the watermark detection procedure[COX02]. For examples, owner identification and copy control systems cannot refer to original data [COX02, KEJA03, WONG03]. Furthermore, the use of informed watermarking may cause to confuse the proof of ownership if an illegal user asserts that he/she is the copyright holder with a corrupt watermarked data as his/her original [CRAV97].

2.4 Several operations for 3-D Surface Meshes

In this subsection, we address several operations to manipulate the 3D surface meshes. Since these operations can be used to interfere with the watermark extraction, they are also called *Attacks* in the filed of digital watermarking. We categorize possible attacks on the watermarked 3D surface meshes by two groups: geometrical attacks and topological attacks.

2.4.1 Geometrical Attacks

Geometrical attacks only changes the location of the vertices of the meshes. Additive noise, uniform coordinate quantization, smoothing and RST (rotation, scaling and translation) attacks are considered as geometerical attacks.

Noise

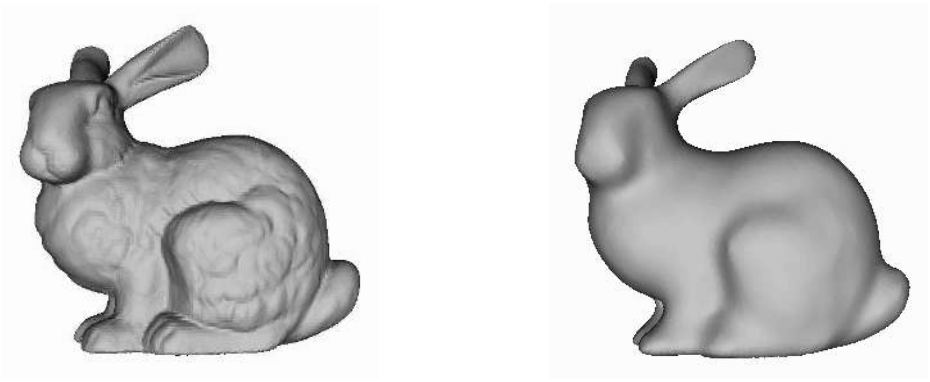
The noise attack is to add random vectors to vertices of the watermarked meshes. This is one of the classical attacks in digital watermarking. Noise can be considered as one of the fashion to modelize geometry compression [CAYR03].

Smoothing

Smoothing operation reduces high frequency information in the geometry of the mesh. In this thesis, we use Laplacian smoothing proposed by Field [FIEL88]. Laplacian smoothing is done by simply averaging neighboring vertices iteratively. Laplacian smoothing eliminates high-frequency noise according to relation factor *lambda* and iteration times. A step of the process is described by the following formula:

$$v_i^{'} = v_i + \lambda \sum_{n=0}^{d(v_i)} (\frac{v_n - v_i}{d(v_i)}) \quad , \tag{2.27}$$

where $0 < \lambda < 1$, v_n is neighbor vertices of v_i and $d(v_i)$ is the valence of vertex v_i. For each vertex $v_i \in V$, the coordinates of v_i are modified according to an average of neighboring vertices of v_i. A relaxation factor λ is available to control the amount of displacement of v_i. With excessive iterations of the smoothing, important details may be lost and the surface may shrink toward the centroid.

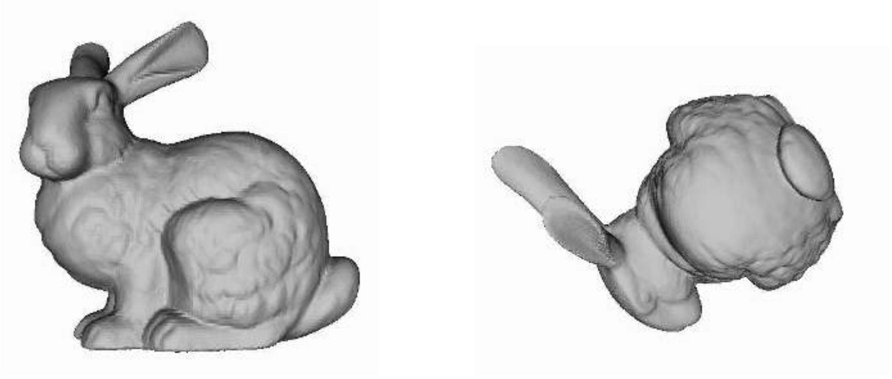


(a) Stanford bunny (b) Smoothed model

Figure 2.15: Example of Laplacian smoothing of 3D model

Similarity transform

Similarity transform (rotation, uniform scaling and translation) is a sub-group of affine transform which consists of a linear transformation followed by a translation. Since the visualization is unrestricted to rotation, uniform scaling and translation of 3-D object, similarity transform does not affect geometrical distortion.

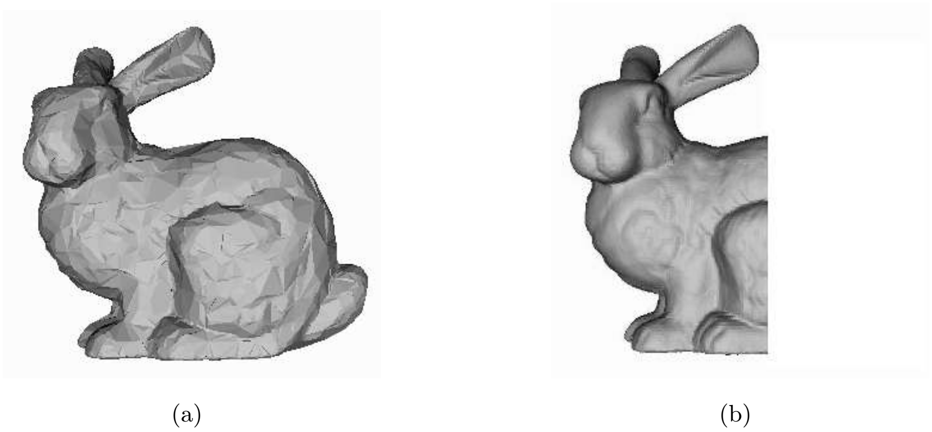


(a) Stanford bunny (b) Affine transformed model

Figure 2.16: Example of affine transform of 3-D model

2.4.2 Topological Attacks

Topological attacks change connectivity information of the 3-D model. Random connectivity re-ordering, simplification, subdivision, re-meshing and cropping are regarded as topological attacks.



(a)

(b)

Figure 2.17: Example of simplification and cropping of 3-D mesh

Random vertex re-ordering

Random re-ordering only changes the connectivity information of the 3-D model, but the location of vertices does not distort. Similar to similarity transform, it does not affect geometrical distortion because the connectivity information is not visible.

Simplification

Simplification is also called decimation [GARL97] because it reduces the number of polygons of the model. It is often used to transmit a low resolution level of 3-D mesh or optimize a model by eliminating most of the non-salient triangles. Simplification of 3-D model can be regarded as down-sampling operation of 2-D image.

Crop

Crop attack reduces the number of vertices and triangles of the model by cutting into several sections. The geometry and connectivity of the remained section after cropping are unchanged.

Re-meshing

Re-meshing is used to regularize a surface meshes by converting an irregular mesh into a semi-regular or a completely-regular one. This operation can be regarded as a geometric re-sampling followed by re-definition of connectivity of the vertices.

Chapter 3

State of The Art

In this chapter, we give a state of the art on the digital watermarking methods for 3-D surface meshes and mesh sequences, which are mainly interested in this thesis. In particular, we explain watermarking methods using the distribution of vertex norms in the spatial domain [CHO04, CHO07] and previous wavelet transform based watermarking methods [KANA98, UCCH04].

3.1 3-D Surface Meshes Watermarking

3-D polygonal mesh models have many difficulties for watermark embedding. While image data is represented by gray-level of pixels sampled over a regular grid in two dimension, 3-D polygonal models have no unique representation, i.e., no implicit order and connectivity of vertices [YU03b]. This yields synchronization problem during the watermark extraction, which makes it difficult to develop robust and blind watermarking techniques. For this reason, most techniques developed for other types of multimedia such as audio, image and video stream are not effective for 3-D meshes. Furthermore, a variety of complex geometrical and topological operations that we mentioned in Section 2.4 could disturb the watermark extraction for ownership assessment.

Since 3-D mesh watermarking techniques were introduced in [OHBU97], there have been several attempts to improve the performance in terms of transparency and robustness. In the first publication on 3-D watermarking, Ohbuchi *et al.* [OHBU97] introduced several watermark embedding primitives in spatial domain. They can be classified into two groups: geometrical components and topological components. Geometrical components consist of geometrical information of surface meshes. Based on coordinates of a vertex, a length of a edge and an area of a polygon can be used to be robust against translation and rotation attack. The angle of two triangles which is invariant to similarity transform belongs to geometrical primitives. Topological components such as two alterative ways of triangulating a quad-lateral are also introduced. Note that, most of watermarking methods

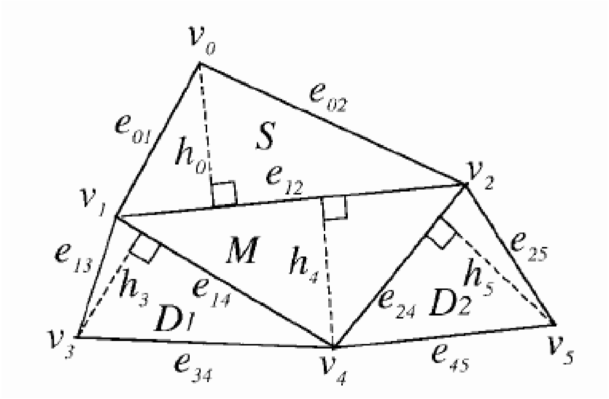


Figure 3.1: A macro embedding primitive by the local arrangement of edges in TSQ method [OHBU98]. v_i are vertices, e_{ij} are lengths of the edges, and h_i are height of the triangles.

embeds the watermark into geometrical components [OHBU98, BENE99, YU03a, YU03b, WU05, CHO04, CHO07], because topological components could be fatally destroyed by the alteration of connectivity, for example, simplification and vertex re-ordering.

Ohbuchi *et al.* [OHBU98] proposed three watermarking schemes: TSQ (Triangle Similarity Quadruple), TVR (Tetrahedral Volume Ratio) and a visible mesh watermarking method. They defined a macro embedding primitive (MEP) based on topological characteristic of the surface meshes before watermark embedding. The MEP can be regarded as a bin which carries a bit of watermark in the our watermarking methods. TSQ and TVR methods uses local arrangement of the edges to embed the watermark. Fig. 3.1 shows an example of macro-embedding primitive(MEP) in TSQ method. After finding a set of four triangles to be used as an MEP, it first changes vertices in Marker triangle, $M \in \{v_1, v_2, v_4\}$ as a ratio of $e_{14}/e24$ and $h_4/e12$ to embed a watermark bit. Then, the remaining three vertices $\{v_0, v_3, v_5\}$ are changed to keep the ratio of the height h_0, h_3 and h_5. Blind detection is available by finding the MEP and the ratio of height in marker triangle but the methods are not sufficiently robust against various attacks. In particular, TSQ and TVR are vulnerable to topological attacks such as re-meshing, simplification and random vertex re-ordering.

Yu *et al.* [YU03a, YU03b] proposed a vertex norm modification method that perturbs the vertex norms according to watermark bit to be embedded. This method employs, before watermark embeddeding, scrambling of vertices for the purpose of bin generation. Note that each bin carries one bit of the watermark. The method showed the robustness against noise, simplification as well as cropping attack. However, their methods do require the original meshes since the exact vertex index order is required in the watermark extraction procedures, because they use vertex perturbation in the watermark embedding for the robustness and the invisibility of watermark.

Statistical features can be promising watermark carriers as they are generally less sensitive to distortion-less attacks such as random vertex re-ordering and similarity transform [CHO07]. Several features can be obtained directly from 3-D meshes, particularly, the distribution of vertex directions [BENE99] and that of vector norms [CHO04, CHO07].

Benedens [BENE99] proposed a watermark embedding method that modifies the local distribution of vertex directions from the center of mass of model. As the distribution of the vertex direction is less sensitive to topological alteration, the method is robust against simplification attack. An extended scheme was also introduced in [LEE03] to overcome a weakness to cropping attack. In addition, although these methods [BENE99, LEE03] do not require the original mesh to extract the watermark, they require the radius of vertices and the original center of mass as a side information for pre-processing. Note that the distribution of the vertex directions essentially varies with the degree of rotation, therefore, the methods are not robust against rotation attack.

3.1.1 Blind Watermarking Methods Using the Distribution of Vertex Norms

Distribution of vertex directions has been used as a watermark carrier [BENE99, LEE03], where vertices are grouped into distinct sets according to their local direction and the distribution of vertex direction is altered in each set separately. The distribution does not change by vertex re-ordering operation, but it varies in essence with rotation operation. Thus, it requires re-orientation processing before watermark detection. The distribution of vertex norms does not change either by vertex re-ordering or rotation operations. In this subsection, we describe our previous works [CHO04, CHO07] which use the distribution of vertex norms in . Note that our proposals introduced in Chapter 5 and 6 use the distribution modification method.

Cho and Kim *et al.* [CHO04]

In our previous work of Cho and Kim *et al.* [CHO04], a blind watermarking scheme was proposed. This method embeds the watermark information by modifying the distribution of vertex norms. Fig. 3.2 depicts the main idea behind the approach. Assume that vertex norms have a uniform distribution in the range of $[0, 1]$. To embed a watermark bit of $+1$, the distribution is modified so that its mean value is greater than a reference value as shown in Fig. 3.2 (b). To embed -1, the distribution is modified so that it is concentrated on the left side, and the mean value becomes smaller than a reference as shown in Fig. 3.2 (c).

To embed N bits of the watermark, all vertex norms are divided into N bins and normalized into the rage of $[0, 1]$. Each bin is used as a unit to embed one bit of the watermark bit. Fig. 3.3 shows the real distribution of a bunny model, which is divided into bins by dashed vertical lines. It also shows that the distribution of each bin is close to

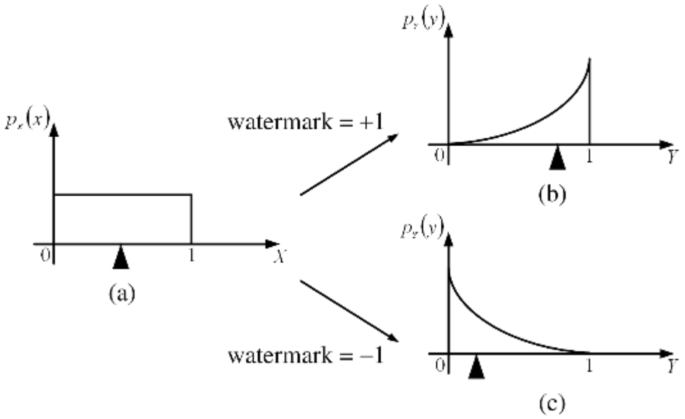


Figure 3.2: Watermarking method by shifting the mean of the distribution in [CHO04] and [CHO07]. Here, dark triangle denotes the mean value in the bin

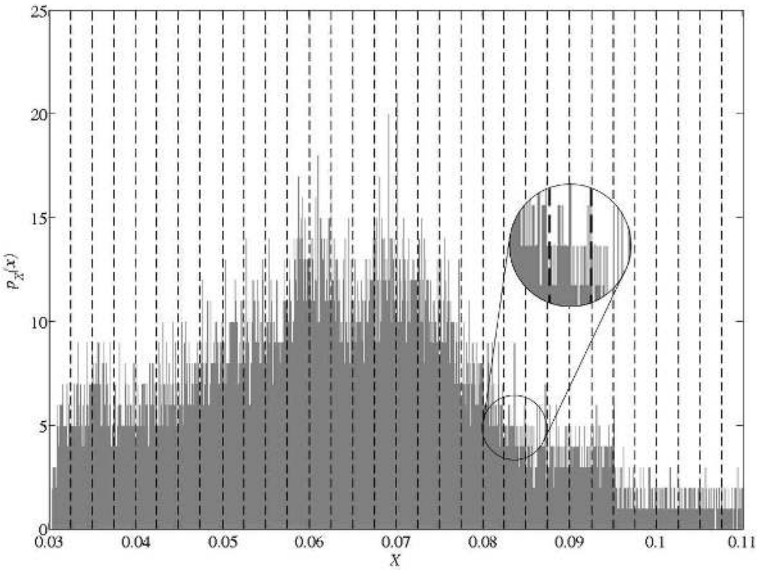


Figure 3.3: Distribution of vertex norms obtained from the bunny model, where dashed vertical lines indicates the border of each bin

uniform. We assume that the vertex norms in each bin have uniform distribution. Then, the distribution is modified according to the watermark bit to be embedded. The mean value μ_n and the reference value r_n is calculated prior to modify the vertex norms in each bin.

$$\mu_n = \frac{1}{M_n} \sum_{j=0}^{M_n-1} \rho_{n,j} \tag{3.1}$$

where M_n is the number of vertices in the n-th bin and $\rho_{n,j}$ is the j-th vertex norm of the n-th bin and

$$r_n = \rho_{\min} + \left(\frac{\rho_{\max} - \rho_{\min}}{N} \right) \cdot n + \frac{1}{2} \cdot \left(\frac{\rho_{\max} - \rho_{\min}}{N} \right) \tag{3.2}$$

where $\rho_{\min}$ and $\rho_{\max}$ are the minimum and maximum vertex norm in the n-th bin, respectively.

To embed the n-th watermark bit, each vertex norm in the n-th bin is modified such that

$$\rho'_{n,j} = \rho_{n,j} + \alpha_n \omega_n \tag{3.3}$$

where ω_n is the watermark bit and α_n is the strength factor of the watermark. α_n is determined so as to guarantee such that $\mu'_n > r_n$ for $\omega_n = +1$ and $\mu'_n < r_n$ for $\omega_n = -1$, respectively.

$$\begin{aligned} &\alpha_n > |r_n - \mu_n| \text{ if } (\omega_n \ = \ +1 \text{ and } \mu_n \ < \ r_n) \ \textit{or} \ (\omega_n \ = \ -1 \text{ and } \mu_n \ > \ r_n) \\ &\alpha_n = 0 \text{ , otherwise} \end{aligned} \tag{3.4}$$

where ${\mu_n}'$ is the new mean value modified by watermark embedding. All vertex norms in each bin are added by $+\alpha_n$ or $-\alpha_n$. The relation between mean and reference values is preserved, when the processing is performed in separate bin by Eq. (3.4). Note that α_n needs to be carefully determined, because the modified vertex norms belong to certain bin could place into other neighbor bins. This artifact could affect the invisibility and the robustness of the watermark [CHO07].

Watermark detection process is quite simple, as the hidden watermark bit can be easily extracted by comparing the mean value of vertex norms and a reference value.

$$\omega''_n = \begin{cases} +1, & \text{if } \mu'_n > r'_n \\ -1, & \text{if } \mu'_n < r'_n \end{cases} \tag{3.5}$$

Although the watermark extraction in the case of cropping attack needs the original center of gravity like in [BENE99], the watermark extraction does not require the original mesh in the most of geometrical and topological attacks such as noise, vertex re-ordering and similarity transform [CHO04]. Note that the method can extract the error-free watermark in the case of random vertex re-ordering and similarity transform.

Cho *et al.* [CHO07]

Cho *et al.* [CHO07] proposed two enhanced methods of their previous work [CHO04]. The distribution of vertex norms are also used for the sake of the robustness against various attacks and blind detection.

In this work, the distribution of vertex norms is modified by two methods. One is to shift the mean value of the vertex norms according to the watermark bit to be embedded and another to change its variance. A similar approach has been used to shift the mean value in our previous work [CHO04], where a constant is added to vertex norms. Note that more sophisticated skills are introduced in this method. In particular, histogram mapping functions are newly introduced and used for the purpose of elaborate modification. Since the statistical features are invariant to distortion-less attacks and less sensitive to various distortion ones with local geometric alterations, robustness of watermark can be easily achieved. In addition, the proposed methods use the blind watermark detection scheme.

To embed N bits of the watermark, the probability distribution of vertex norms are divided into N bins according to their magnitude. Each bin is normalized into the range of $[0, 1]$. Assume that each bin has a uniform distribution over the interval of $[0, 1]$, then the mean value of each bin is modified according to the watermark as shown in Fig. 3.2. In contrast to additive watermark embedding in [CHO04], histogram mapping function was proposed to shift the mean value. Histogram mapping functions can effectively modify the distribution and reduce the visibility of the watermark as much as possible. The mapping function is given by

$$Y = X^k \text{ for } 0 < k < \infty \text{ and } k \in \Re \tag{3.6}$$

where X is a continuous random value with uniform distribution over the interval $[0, 1]$, Y is the transformed variable, and the parameter k is a real value for $0 < k < \infty$. Clearly, the expectation of the random variable $E\left[X\right]$ is given by

$$E\left[X\right] = \int_0^1 x p_X\left(x\right) dx = \frac{1}{2} \tag{3.7}$$

where $p_X\left(x\right)$ is the PDF (Probability Density Function) of X. This expectation $E\left[X\right]$ is used as a reference value when moving the mean of each bin to a certain level. Vertex norms in each bin are modified to shift the mean value. It is very important to assure that the modified vector norms also exist within the range of each bin. Otherwise, vector norms belonging to a certain bin could shift into neighbor bins, which may have a serious impact on the watermark extraction [CHO04]. In [CHO07], histogram mapping function of Eq. (3.6) can shift the mean to the desired level through modifying the value of vector norms while staying within the proper range.

Fig. 3.4 shows curves of the mapping function for different values of k. When the parameter k is selected in the range $1 < k < \infty$, input variables are mapped into relatively small values. Moreover, increases in k decrease the value of the transformed variable. It means the decrease of mean value. On other hand, the mean value increases for decreasing k when $0 < k < 1$. Expectation of output random variable, $E[Y]$, is represented as;

$$E\left[Y\right] = E\left[X^k\right] = \int_0^1 x^k p_X\left(x\right) dx = \frac{1}{k+1}. \tag{3.8}$$

Fig. 3.5 shows the expectation of the output of the mapping function over k. The expectation value decreases monotonically with the parameter k. Therefore, we can easily adjust the mean value of the distribution by selecting a proper parameter. In particular, the mapping function does not only guarantee to alter the variable within the limited range, but also allows shifting of the mean value to the desired level.

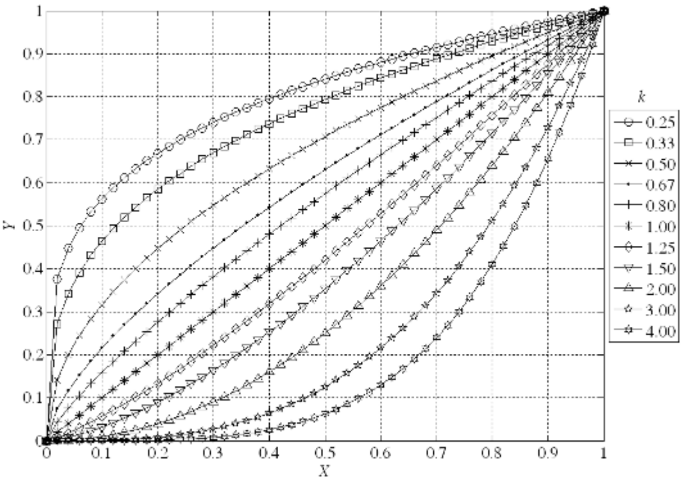


Figure 3.4: Histogram mapping function, $Y = X^k$, for different parameters of k

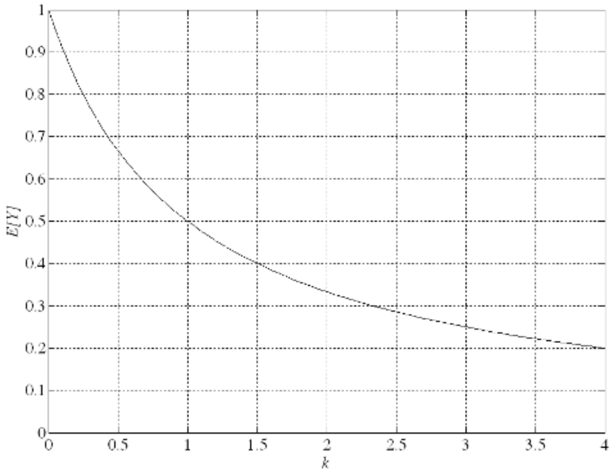


Figure 3.5: Expectation of the output random variable via histogram mapping function with different k, assuming that the input random variable is uniformly distributed over unit range $[0, 1]$.

The watermark extraction is similar to [CHO04]. The vertex norms of the watermarked mesh are classified into N bins and normalized into the range of $[0, 1]$. The mean of each bin, μ_n, is calculated and compared to the reference value, $\frac{1}{2}$ by means of

$$\omega_n = \begin{cases} +1, & \text{if } \mu_n > 1/2 \\ -1, & \text{if } \mu_n < 1/2 \end{cases} \tag{3.9}$$

where ω_n denotes the watermark in n-th bin.

From the main idea of distribution and histogram mapping function, variance shifting method was also proposed in [CHO07]. Fig. 3.6 shows the main idea of the variance shifting method. To embed N bits of the watermark, the probability distribution of vertex norms are divided into N bins according to their magnitude. Each bin is now normalized

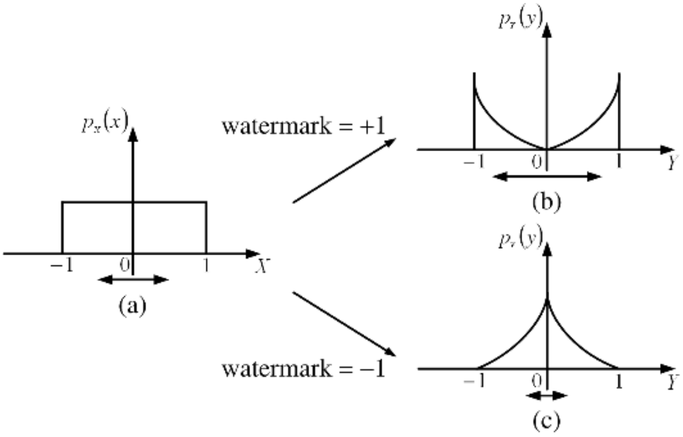


Figure 3.6: Watermarking method by changing the variance of the distribution in [CHO07]. Here, the standard deviation of the bin is indicated by bi-directional arrow.

into the range of $[-1, 1]$. Assume that each bin has a uniform distribution over the interval $[-1, 1]$ as shown in Fig. 3.6 (a). To embed a watermark bit of $+1$, the distribution is modified to concentrate on both margins. This leads to increase the standard deviation as shown in Fig. 3.6 (b). To embed -1, the distribution is altered to concentrate on the center so that the standard deviation becomes smaller than a reference deviation as shown in Fig. 3.6 (c). The mapping function for the variance changing method is defined by;

$$Y = sign\left(X\right)\left|X\right|^{k} \text{ for } 0 < k < \infty \text{ and } k \in \Re \tag{3.10}$$

where X is a continuous random variable with uniform distribution over $[-1, 1]$, Y is the transformed variable, and k is a real value for $0 < k < \infty$. As X has a zero mean, its variance is given by;

$$E\left[X^2\right] = \int_{-1}^{1} x^2 p_X\left(x\right) dx = \frac{1}{3} \tag{3.11}$$

where $E[X^2]$ denotes the second moment of the random variable X. A variance of $\frac{1}{3}$ is used as a reference when changing the variance of each bin according to the watermark bit to be embedded. To change the variance, vector norms in each bin should be modified within the normalized range of $[-1, 1]$. For this purpose, we use a histogram mapping function, which can change the variance to the desired level by modifying vector norms while staying within the specified range.

Fig. 3.7 shows curves of the mapping function for different k. When the parameter is selected for $1 < k < \infty$, input variable is transformed into output variable with relatively small absolute value while maintaining its sign. Moreover, the absolute value of transformed variable becomes smaller as increasing k. It means a reduction of the variance. On the other hand, variance increases for decreasing k on the range $0 < k < 1$. The variance of

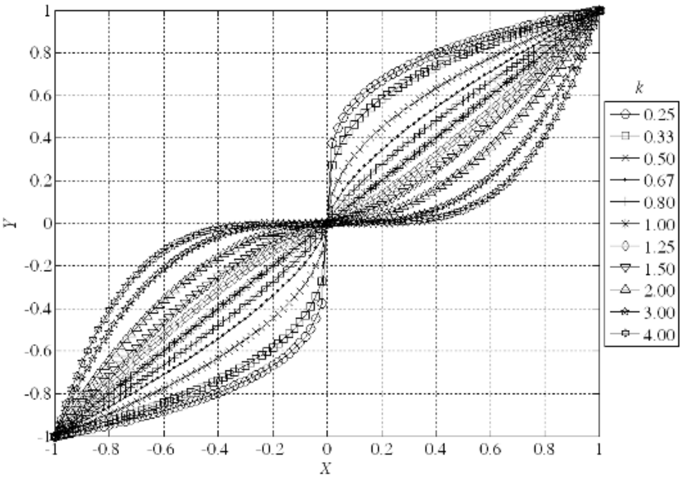


Figure 3.7: Histogram mapping function, $sign\left(X\right)\left|X\right|^{k}$, for different parameter of k

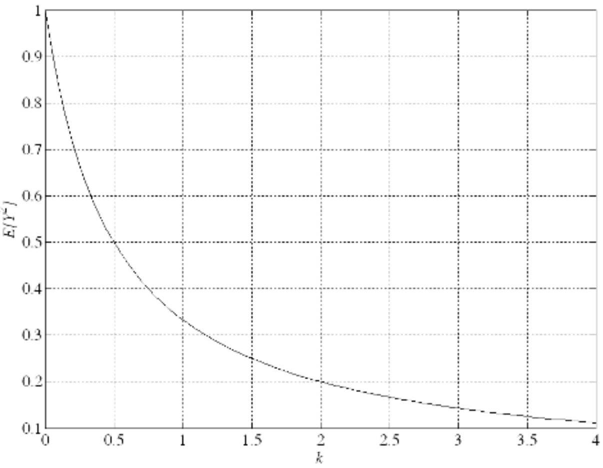


Figure 3.8: Variance of the output random variable via histogram mapping function with different k, assuming that the input random variable is uniformly distributed over the normalized range $[-1, 1]$

the output random variable $E\left[Y^2\right]$ is represented as

$$E\left[Y^2\right] = E\left[\left(sign\left(X\right)\left|X\right|^k\right)^2\right] = \int_{-1}^{1} |x|^{2k} p_X\left(x\right) dx = \frac{1}{2k+1} \tag{3.12}$$

Fig. 3.8 shows the variance of the output random variable over k of the mapping function. Note that the variance decreases monotonically as k increases. Therefore, the variance of the distribution can easily be adjusted by selecting a proper parameter.

Similar to mean shifting method, the watermark can be extracted by comparing the reference variance and variance taken from watermarked meshes. The vertex norms of the watermarked mesh are classified into N bins and normalized into the range of $[-1, 1]$. The variance of each bin, μ_n, is calculated and compared to the reference value, $\frac{1}{3}$ by means of

$$\omega_n = \begin{cases} +1, & \text{if } \sigma_n^2 > 1/3 \\ -1, & \text{if } \sigma_n^2 < 1/3 \end{cases} \tag{3.13}$$

where ω_n denotes the watermark in n-th bin.

Both the mean modifying and the variance shifting methods can extract the error-free watermark after vertex re-ordering and similarity transforms. The methods are also fairly robust against noise, smoothing and uniform vertex coordinates quantization attacks as well as simplification and subdivision attacks. However, the robustness against cropping and the invisibility of the watermark is still needs to be enhanced.

3.1.2 Wavelet Transform Based Watermarking Methods

With spatial domain based methods [OHBU97, OHBU98, BENE99, LEE03, YU03a, YU03b, CHO04, CHO07], some frequency domain based watermarking methods have also been introduced [OHBU01, KANA98, YIN01, UCCH04]. Although frequency domain based watermarking methods need more computational complexity for the watermark embedding and extraction process, several advantages can be taken by embedding the watermark into a particular sub-band signal to satisfy the several requirements as we mentioned in Section 2.3.2 and 2.3.3. In this subsection, we describe two wavelet domain based watermarking methods [KANA98, UCCH04] which embed the watermark into the wavelet coefficients, especially into the high frequency band.

Wavelet transform gives several advantages in terms of the watermark embedding primitive. From the view point of watermark invisibility, it causes unnecessarily quality distortion to embed watermark into spatial domain, because the watermark embedder alters whole frequency signal. On the other hand, by using frequency domain based method, we can determine a proper frequency band to embed the watermark. When the watermark is embedded into the wavelet (high frequency) coefficients, the invisibility of the watermark is increased. The robustness can be increased by modifying the scale (low frequency) coefficients. In addition, the capacity of the watermark can be increased by embedding into the wavelet (or scale) coefficients at each resolution level. Note that, the capacity of the watermark at each level might be limited since the number of resolution levels is dependent on the number of vertices of original mesh.

Kanai *et al.* [KANA98]

In 1998, Kanai *et al.* [KANA98] first introduced wavelet analysis based watermarking method for 3D surface meshes. The method employed regular wavelet transform [LOUN94] for multiresolution analysis. The watermark is embedded into L2 norm of wavelet coefficient

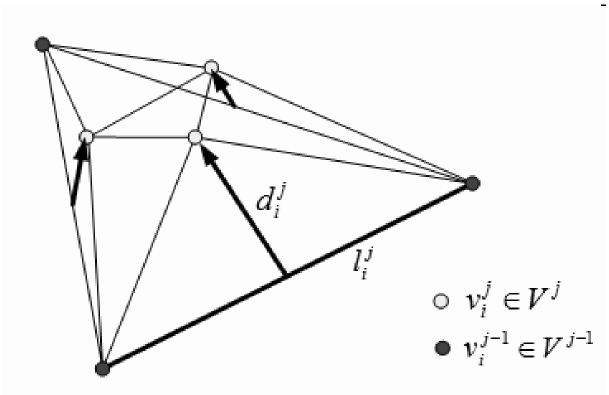


Figure 3.9: Wavelet coefficients d_i^j and the length of the two parent vertices l used in [KANA98]

shown in Fig. 3.9 by spread spectrum method [COX97].

$$d'^j_i = ((\alpha_i^j + \omega_i^j)l_i^j)d_i^j/||d_i^j|| \tag{3.14}$$

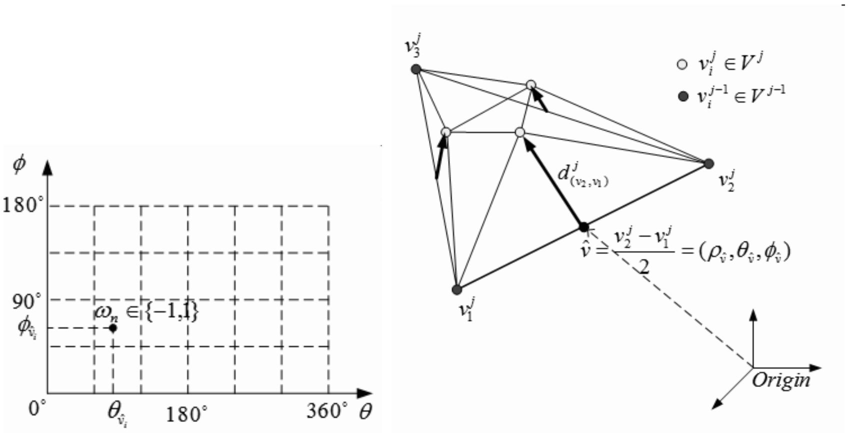
where $||\cdot||$ is the L2 norm, d_i^j is the i-th wavelet coefficients at the j-th level, l is the euclidean distance between the two parent vertices. Here, the strength factor of the watermark, α_i^j varies at each resolution level since the magnitude of L2 norm of wavelet coefficients varies at each resolution level. The watermark ω_i^j is embedded at various resolution levels to achieve high capacity of watermark. An error-free detection is possible for the rotation and translation attacks, as L2 norm of wavelet coefficients is invariant to rotation and translation. One of drawbacks of this method is that the watermark extraction requires the original mesh. Moreover, the input model for watermarking is limited to regular meshes that have 1-to-4 subdivision connectivity since their method employed regular subdivision (Fig. 2.7) for wavelet analysis,. Although they insisted that it can be applied to irregular meshes by using re-meshing technique [VORS01, SURA03, ALLI03], the re-meshed model has a different topology (connectivity) from that of the original. Note that, in our watermarking methods for 3-D surface meshes in Chapter 4 and 5, the watermark extraction does not require the original meshes, moreover, the watermark can be directly embedded into irregular meshes by using the irregular wavelet analysis introduced in Section 2.2.

Uccheddu *et al.* [UCCH04]

Uccheddu *et al.* [UCCH04] extended previous watermarking method [KANA98] to blind detection. The method also modified the L2 norm of wavelet coefficients. In contrast to [KANA98], watermarking map, namely W_{MAP} is generated by using the direction of the application vertex which is used to calculate the wavelet coefficients. Fig. 3.10 depicts the relation between watermarking map W_{MAP} and an application vertex. Since the direction of the application vertex can be changed by rotation attack, the input model needs to be realigned before watermark embedding and extraction. Therefore, The method uses principal component analysis (PCA) as a pre-processing in the watermark embedding and extraction procedures. After the vertex coordinates of the original meshes is normalized to a unit cube of dimensions 1.0 × 1.0 × 1.0, PCA is applied by five calculating the covariance matrix as follows:

$$I = \begin{bmatrix} \sum_{i=1}^{n} x_i^2 m_i & \sum_{i=1}^{n} y_i x_i m_i & \sum_{i=1}^{n} z_i x_i m_i \\ \sum_{i=1}^{n} x_i y_i m_i & \sum_{i=1}^{n} y_i^2 m_i & \sum_{i=1}^{n} z_i y_i m_i \\ \sum_{i=1}^{n} x_i z_i m_i & \sum_{i=1}^{n} y_i z_i m_i & \sum_{i=1}^{n} z_i^2 m_i \end{bmatrix} \tag{3.15}$$

where $0 \leq i < |V|$, (x_i, y_i, z_i) are the coordinates of the vertex which are relative to the center of gravity of the mesh. The eigenvectors of I are computed and used to align the principal axis of I to the coordinate axis by multiplying the location of each vertex by an appropriate matrix. PCA is also applied in spatial domain based watermarking methods [KALI03, ZAFE05] for the blind watermarking.



(a) Watermarking map, W_{MAP}

(b) The application vertex, $\hat{v}$

Figure 3.10: Watermarking map, W_{MAP} and the application vertex, $\hat{v}$in [UCCH04]

The watermark signal ω_n is uniformly distributed in the $[-1, 1]$ interval where $0 \leq n < |N|$, $|N|$ is the number of watermark signal. Then, ω_n is arranged into a matrix with r rows and c columns ($N = r \times c$). W_{MAP} is a mapping of the watermark signal ω_n into polar coordinates, where the row index of W_{MAP} samples azimuth angle ($\theta \in [0, 2\pi]$) and the column index samples the elevation angle ($\phi \in \pi$)).

The n-th watermark signal is embedded into the whole L2 norms of wavelet coefficients that belong to n-th W_{MAP} by,

$$||d'_{n,j}|| = ||d_{n,j}|| + \alpha\omega_n(\hat{v}(d_{n,j}), \theta_{\hat{v}}, \phi_{\hat{v}}) \tag{3.16}$$

where $0 \leq j < |D|$, $||.||$ denotes the L2 norm and α is the strength factor that controls the invisibility and the robustness of the watermark. Note that, the watermark signal of n-th

W_{MAP} is determined by the direction of the application vertex $\hat{v}$. Fig. 3.10-(b) depicts an application vertex $\hat{v}$, where $\hat{v}(d_{v2,v1}) = \frac{v_2 - v_1}{2}$, v_1 and v_2 are two parent vertices to calculate the wavelet coefficient, respectively.

Since the method embeds an 1-bit watermark, also called detectable watermark, the watermark detector only tells the presence of the watermark. To detect the watermark, the wavelet coefficients are obtained from the watermarked meshes, and the correlation between the designed watermark signal and the whole wavelet coefficients in n-th W_{MAP} [COX97, COX02]. Similar extraction method based on the correlation is used in our method in Chapter 4, but our method embed multi-bit watermark, also called readable watermark.

By using PCA before the watermark embedding and extraction procedure, the method showed fairly robustness against rotation and translation. One of the drawbacks of the approach is that, the input mesh is still limited to regular meshes as is in [KANA98]. The method is fairly robust against additive noise and smoothing. However, it might be vulnerable to topological attacks that cause a synchronization problem. The exact W_{MAP} by using the wavelet coefficients and the application vertex cannot be guaranteed after topology attacks. Note that, we explain the synchronization problem in Chapter 4 and 5.

3.2 3-D Mesh Sequence Watermarking

In this section, we address digital watermarking for 3-D mesh sequence. We also introduce temporal wavelet transform based watermarking methods for video data [SWAN98a] and motion data [KIMT00, YAMA04].

Unlike key-frame animation is used for real-time animation of 3-D medels in VRML (Virtual Reality Modeling Language) file format, mesh sequences are utilized to represent realistic visual data in many applications such as character animation in movies or video games, cartoon, 3-D medical diagnosis, and so on. The enormous data size of 3-D mesh sequence has brought many researches on its effective representation and the compression [LENG99, GUSK04, PAYA05, IBAR03, CHO06]. Although watermarking techniques have been developed for 3-D static meshes, there has been no watermarking methods for 3-D mesh sequences based to our knowledge even though 3-D surface meshes models and 3-D mesh sequence have already been commercially available. Obviously, 3-D mesh sequences are also needed to be protected as copyrighted contents, as it is very time-consuming and high effort work to create or design such 3-D mesh sequences.

The watermarking for mesh sequences needs to consider the robustness against several attacks as well as the invisibility and capacity of the watermark like in 3-D static mesh watermarking. For instance, several attacks in 3-D static meshes watermarking, namely intra-frame attacks, are possible to each frame of the mesh sequence. 3-D mesh sequence watermarking has another difficulty that generally do not have a counterpart in the watermarking of 3-D static meshes. Because of temporal redundancy of consecutive frames, mesh sequence are exposed to inter-frame attacks, such as frame averaging, frame dropping, frame swapping and so on. These attacks may visually affect little or no damage to the sequence while the watermark in the sequence may be destroyed. Clearly, the watermarking of 3-D mesh sequence should consider the robustness against both intra-frame and inter-frame attacks.

For the robustness of the watermark, we propose 3-D mesh sequence watermarking methods based on the temporal wavelet transform. in Chapter 6. The temporal wavelet transform have often used in video watermarking methods [SWAN98a, KIM02, LIYI99].

In 1998, Swanson *et al.* [SWAN98a] introduced a watermarking method for copyright protection of video data based on temporal wavelet transform. Prior to embedding process, the original video sequence that would be watermarked was segmented into scenes and temporal wavelet transform is applied to each scene, then a scene is decomposed into temporal low frequency and high frequency frames. The watermark is embedded into 8×8 DCT transformed coefficients of both low and high frequency frames. The correlation between the designed watermark signal and the watermarked video data is calculated for the watermark detection. The method also showed the possibility of the watermark detection from a single frame. This is a useful when we have no knowledge of the frame index of video sequence. The experimental results showed the robustness against various attacks such as additive noise, MPEG video compression and frame dropping. Note that the method embeds the detectable (1-bit) watermark and the correlation shows only the presence of the watermark similar to in [HART98, UCCH04]. 1-bit of the watermark is not sufficient to contain the information of the copy-right holder. It is evident that readable (multi-bit) watermark is more useful and required for the application of copy-right protection. For instance, Digital Cinema System [DCI, CIFE] specifies minimum 35 bits of capacity for every 5 minutes to embed 16 bits of time stamp (time, date and year) and 19 bits of serial number to indicate location of the theater.

There have been two watermarking methods for the motion data for articulated computer graphics figures (e.g. the movement of joints in human body over the time) [KIMT00, YAMA04]. In general, the motion signal is obtained by motion capture devices and should be associated with the 3-D model to be animated. In contrast to the data structure of mesh sequences, the motion data which is used in [KIMT00, YAMA04] consists of the angles of the joints, the marker positions, and the orientation of a body segment [YAMA04, KIMT00]. In their methods, the motion signal is decomposed into frequency domain, and the watermark is embedded into the transformed coefficients of high (or middle) frequency domain by using spread-spectrum method [COX97]. The watermark detection requires the pre-processing such as registration and re-sampling using the original motion signal. Note that these methods are only applicable to a special animation method which uses the motion signal.

Part I

Wavelet Analysis based Digital Watermarking for 3-D Surface Meshes

Chapter 4

Digital Watermarking Using Wavelet Coefficient Norms

In this chapter, we introduce a digital watermarking method for 3-D surface meshes that embeds the watermark into the wavelet coefficients. The proposed method is designed for blind watermark detection, which can extract the watermark without reference of cover mesh model. To ensure the connectivity synchronization after inverse wavelet transform, we introduced a vertex and face re-ordering process as pre-processing in both watermark embedding and extraction. Since the wavelet coefficient norms is invariant to translation and rotations, the method is also robust against rotation and translation. This work has published in [KIM05].

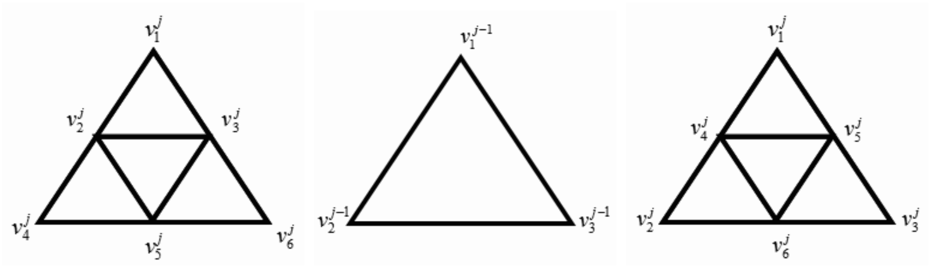
4.1 Introduction

In the previous works, Kanai *et al.* [KANA98] proposed a watermarking method to embed the watermark into the wavelet coefficient vector in a certain order. The method requires the original mesh for the topological synchronization of the vertices. Uccheddu *et al.* [UCCH04] introduced a blind detection scheme by using the direction of application vertex, but this method embeds 1-bit of detectable watermark and it is just applicable to the input meshes with regular subdivision connectivity as was in [KANA98]. In contrast with the previous approaches [KANA98, UCCH04], as the irregular wavelet analysis [VALE04a] supports 1-to-4, 1-to-3, 1-to-2 and 1-to-1 subdivisions, our proposed method is directly applicable to both irregular and regular meshes.

4.2 Blind Watermarking Using Wavelet Coefficients Norms

4.2.1 Preprocessing for wavelet transform of 3-D surface meshes

To apply both irregular and regular meshes to watermarking, there is still an important problem to be solved. Unlike discrete wavelet transform (DWT) of 2-D images, the wavelet coefficients of 3-D meshes depend on the seed-triangle group merged during connectivity graph simplification. This is mainly caused by that the wavelet synthesis is not designed to recover the exact topological information. Fig. 4.1 shows an example of connectivity synchronization problem. The vertex index of reconstructed meshes $\hat{M}^j$ (Fig. 4.1-(c)) is different with that of the original meshes M^j. Although the lossless reconstruction of integer coordinates meshes is possible and the reconstructed mesh $\hat{M}^j$ from M^0 is exactly the same as the original meshes M^j in terms of geometry, the original connectivity is changed when processing with both the regular wavelet analysis [LOUN94] and the irregular wavelet analysis [VALE04a]. Note that the variation of connectivity affects the seed-triangle group and seed-vertex of the reconstructed meshes.

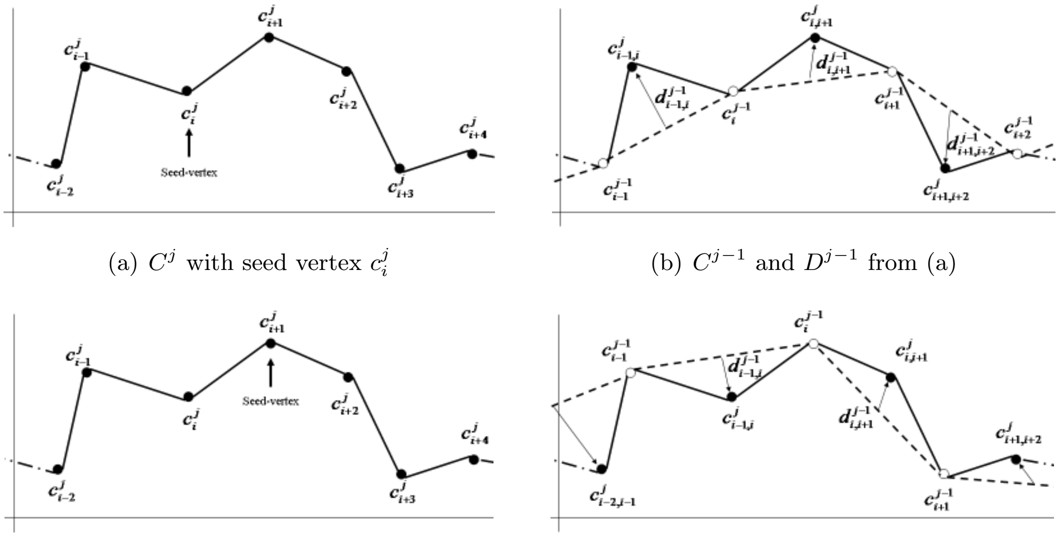


(a) M^j (b) M^{j-1} (c) $\hat{M}^j$

Figure 4.1: Synchronization problem by subdivision of surface meshes: (a) the original meshes M^j, (b) simplified mesh M^{j-1} from M^j, and (c) the reconstructed meshes $\hat{M}^j$.

The different seed-cell and seed-vertex produces completely different wavelet coefficients. Fig. 4.2 shows another example of the synchronization of wavelet coefficients. The wavelet coefficients of Fig. 4.2-(b) are calculated by the seed-vertex of Fig. 4.2-(a) and the wavelet coefficient of Fig. 4.2-(d) are calculated by Fig. 4.2-(c). Two wavelet coefficients in Fig. 4.2-(b) and (d) are completely different each other. It is clear that the synchronization of wavelet coefficients depends on the seed-triangle group and seed-vertex.

Since Kanai *et al.* [KANA98] extract the watermark by non-blind watermark detection scheme, synchronization problem can be solved by the re-sampling or re-orientation using the original mesh. Uccheddu *et al.* [UCCH04] needs to solve the problem because the watermarked mesh must have the same topological information as that of the original to calculate accurate direction of the application vertex ($\hat{v}$) and its wavelet coefficient. However, Uccheddu *et al.* [UCCH04] have not mentioned the means to recover the exact topological information in the process of watermark detection. Note that the artifact oc-



(a) C^j with seed vertex c_i^j

(b) C^{j-1} and D^{j-1} from (a)

(c) C^j with seed vertex c_{i+1}^j

(d) C^{j-1} and D^{j-1} from (c)

Figure 4.2: The wavelet coefficients vector with different seed-vertex. The different seed-cell and seed-vertex destroy the synchronization of wavelet coefficient vector

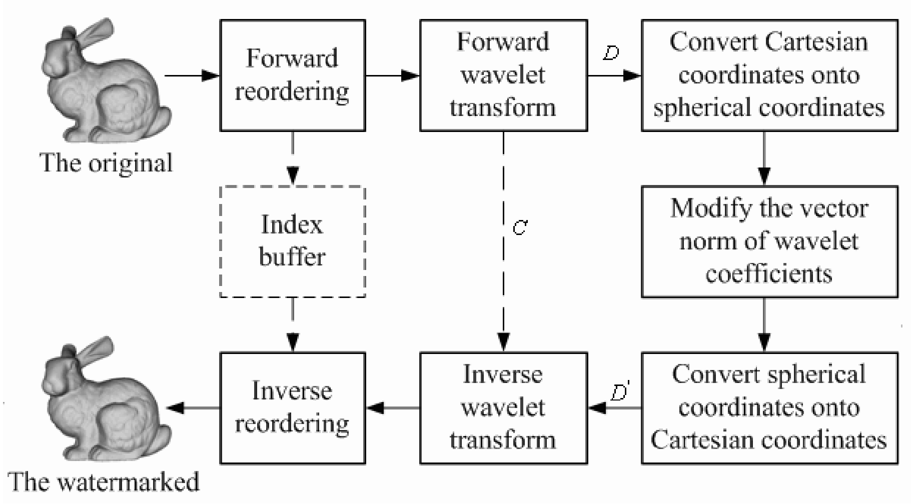

The original
Forward reordering
Forward wavelet transform
D
Convert Cartesian coordinates onto spherical coordinates
Index buffer
C
Modify the vector norm of wavelet coefficients
The watermarked
Inverse reordering
Inverse wavelet transform
D'
Convert spherical coordinates onto Cartesian coordinates

Figure 4.3: A block diagram of watermark embedding procedures

curred from inaccurate topological when the wavelet coefficients are used for the watermark embedding primitives.

In this proposal, we use a pre-processing to recover the original topology (connectivity) by using re-ordering algorithm. The block diagram of the watermark embedding procedures are shown in Fig. 4.3. Before we perform forward wavelet transform, we re-order the connectivity information of the input surface meshes. During the re-ordering, the original connectivity information is stored in the buffer. This buffer will be used to recover the original topology after inverse wavelet transform. Our walking algorithm for re-ordering an irregular mesh is inspired both from Touma-Gotsman connectivity coder [TOUM98] and well-known backtracking algorithm [TOUM98] for the visit of tree nodes.

The re-ordering starts from a seed triangle which is determined from geometry criteria such as the triangle with the largest area and the farthest triangle from the model centroid. Connectivity information, for example, the triangle which has the maximum number of neighboring triangles within its n-ring, also could be good to be used. Note that, both geometrical and topological criteria to find the seed triangle do not always match with all kinds of input meshes. For example, the valence of n-ring neighbor vertices can be used to process the irregular or semi-regular meshes, but can not be used for regular meshes. It is also very important to select the criterion which is less sensitive to various attacks. After having processed the re-ordered mesh by watermark embedder, the connectivity of resulting mesh is returned by inverse re-ordering process. Re-ordering brings an additional computational cost, however, it enables a wavelet-based watermarking algorithm to have blind detection.

In our proposal, we use a connectivity criterion: the number of neighboring vertices within their n-ring for irregular meshes and semi-regular meshes.

Fig. 4.4 describes our walking algorithm for re-ordering an irregular mesh. From the seed-face $f1$, we define both focus vertex $v1$ and walking vertex $v3$ with the same criterion for the seed-face $f1$ (e.g. the maximum number of neighboring vertices within their 4-ring). Here, the first focus vertex should not have a boundary edge. All the visited face and edge indices need to be stored in index buffers to process the exception handling, for example, a visited face and a boundary edge. Then, we walk to the neighbor face $f2$ which shares $v1$ and $v2$ in $f1$ and store $f2$ and the edge $(v1, v2)$ (see Fig. 4.4(a)). The next vertex $v4$ will be the next walking vertex. $f3$ can be found with the same rule with the focus vertex $v1$ and the new walking vertex $v4$ (Fig. 4.4(b), Fig. 4.4(c)). If the neighbor face has already been visited as shown in Fig. 4.4(d), we need to back-trace the last ordered face $f4$ in the index buffer. For back-tracing the buffer, we also need to switch the focus and the walking vertices. Boundary edges also can be treated by the same way as visited faces (Fig. 4.4(e)). All the 2-manifold meshes can be processed by our walking algorithm.

4.2.2 Watermark Embedding and Extraction by Spread Spectrum Method

After we compute forward wavelet transform of the original mesh M, we obtain the wavelet coefficient vector D. Following Eq. (2.21),

$$D^{j-1} = [\ d_0,\ \ldots,\ d_i,\ d_{|D|-1}\ ]^T, \tag{4.1}$$

where, $d_i = [x_i\ y_i\ z_i]^T$ and $|D|$ is the number of wavelet coefficients. We consider the resolution level is $j-1$ for the sake of notational simplicity. We convert the Cartesians

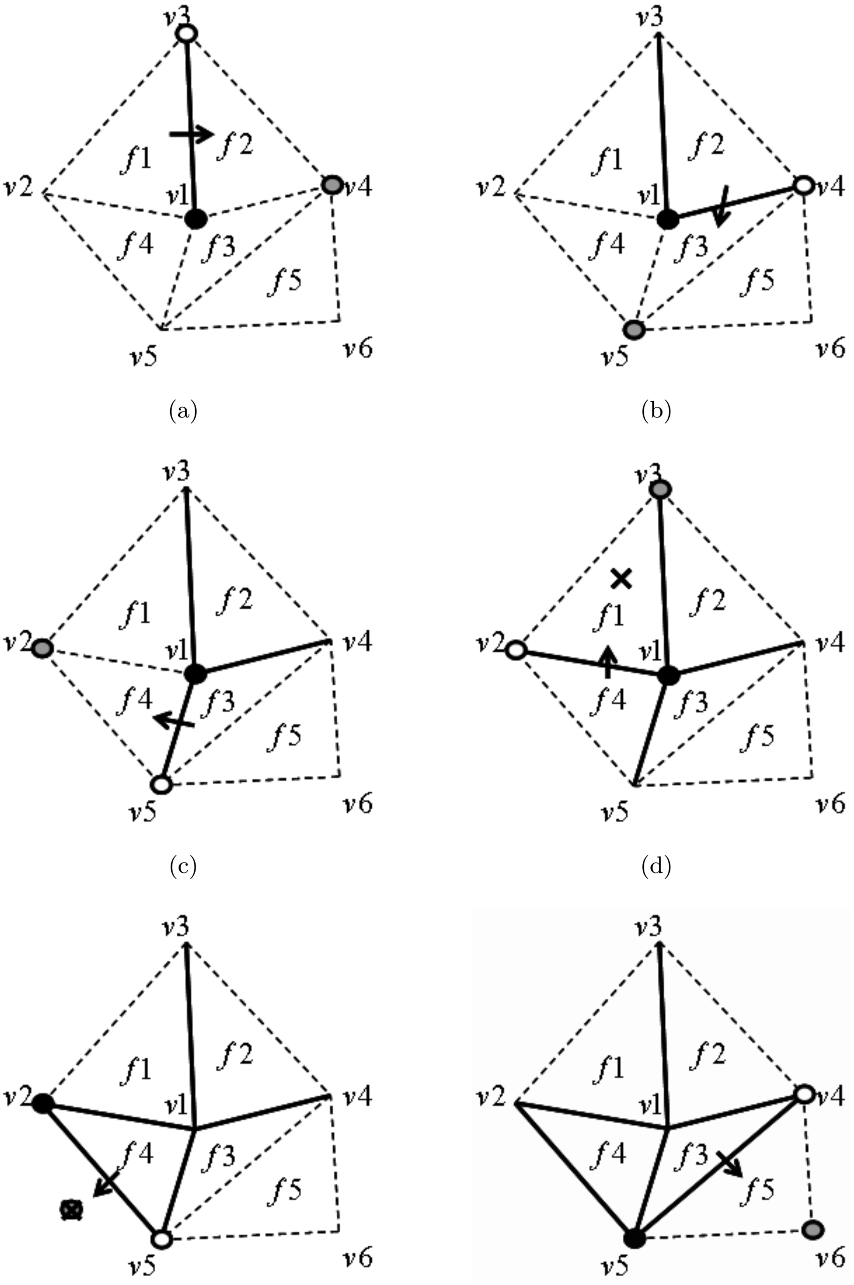


Figure 4.4: Our walking algorithm for re-ordering where dark point is the focus vertex, white point is a walking vertex and gray point is the next vertex. The arrow means the direction to walk. Solid line depicts visited edges and dashed line is unvisited edges.

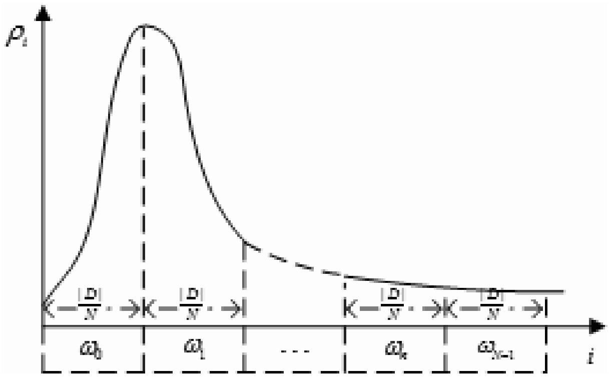


Figure 4.5: The bin generation of the proposed method using the wavelet coefficient norms ρ_i. Each bin carries one bit of the watermark ω_n.

coordinates of the coefficient d_i to spherical coordinates $[\ \rho_i,\ \theta_i,\ \phi_i\ ]^T$ where, ρ_i denotes the i-th wavelet coefficient norm. Note that ρ_i is invariant to rotation and translation attacks.

To embed N bits of the watermark, the wavelet coefficients are divided to N distinct bins as shown in Fig. 4.5. Each bin carries one bit of the watermark. For the robustness of the watermark, the watermark is spread by a zero mean pseudo random binary sequence η_r $(\in \{-1, 1\})$ where $0 \leq r < |D|/N$. Note that the length of pseudo random sequence varies by the number of watermark. The robustness of the watermark can be controlled by the length of pseudo random sequence in this approach. Then, the n-th watermark bit $\omega_n (\in \{-1, 1\})$ modifies the wavelet coefficient norms by means of

$$\rho'_{n,r} = \rho_{n,r}(1 + \alpha\ \omega_n\ \eta_r) \tag{4.2}$$

where r is the index of the wavelet coefficient norms that belongs to the n-th bin, $\alpha(> 0)$ is strength factor of the the watermark. The strength factor can help to increase the robustness of the watermark, but it has to be carefully selected because it also affects the visual quality after embedding the watermark.

Spherical coordinates $[\rho'_i,\ \theta_i\ \phi_i\ ]^T$ are changed into Cartesian coordinates $[x'_i,\ y'_i,\ z'_i]^t$. Finally, the watermarked mesh M' is reconstructed by inverse wavelet transform with the watermarked wavelet coefficients $\tilde{D}$. Note that, before the inverse wavelet transform, the inverse re-ordering is required to recover the connectivity of original meshes as we mentioned in Section 4.2.1.

The watermark extraction is quite simple. Fig. 4.6 shows a block diagram of watermark extraction procedures. Similar to the watermark embedding procedures, forward connectivity re-ordering and wavelet transform of the watermarked mesh M' is required. The watermarked wavelet coefficients d'_i in Cartesian coordinates are converted to spherical coordinates to obtain the wavelet coefficient norms ρ'_i. The wavelet coefficient norms are divided to N distinct bins. We use a blind extraction scheme based on the correlation. To extract the n-th watermark bit, the correlation between $\rho'_{n,r}$ and η_r is calculated by means

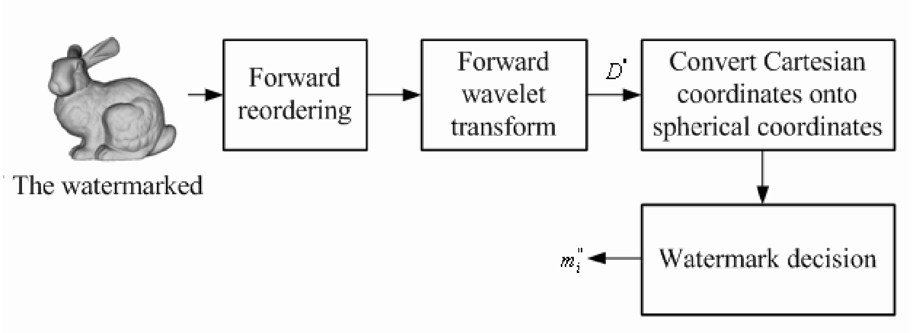

The watermarked
Forward reordering
Forward wavelet transform
D'
Convert Cartesian coordinates onto spherical coordinates
Watermark decision
m_i''

Figure 4.6: A block diagram of watermark extraction procedures

of

$$corr(\rho_n^{'}, \eta) = \sum_r \rho_{n,r}^{'} \; \eta_r = \sum_r \rho_{n,r} \; \eta_r + \alpha \sum_r \rho_{n,r} \; \omega_n \; \eta_r^2 \tag{4.3}$$

Due to the zero mean pseudo random binary sequence η_r, the first term of Eq. (4.3) approaches to zero. In addition, η_r^2 in the second term must be one. Then, the hidden watermark in the n-th bin is determined by

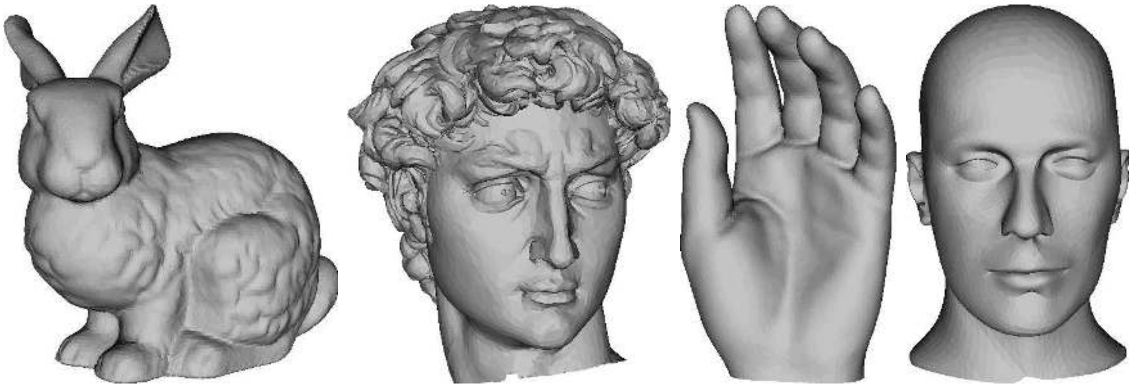
$$\omega_n = \begin{cases} +1, \text{ if } \sum_r \rho_{n,r}^{'} \; \eta_r > 0 \\ -1, \text{ if } \sum_r \rho_{n,r}^{'} \; \eta_r < 0 \end{cases} \tag{4.4}$$

4.3 Experimental Results

In this section, we explain the experimental results to demonstrate the effectiveness of our proposal. Since wavelet based approach is specific of the synchronization of the connectivity, we do not consider the topological attacks such as re-triangulation, simplification or remeshing [UCCH04]. We use four meshes in Table 4.1 and Fig. 4.3. Three of them are irregular meshes: ‘Stanford Bunny’, ‘Davidhead’, ‘Hand’, and the other is a semi-regular mesh, ‘Head’.

Table 4.1: Test Models

	Bunny	Davidhead	Hand	Head
Number of vertices	34,834	20,485	10,196	6,737
Number of faces	69,451	47,753	20,261	13,408
Number of edges	104,288	71,837	30,455	20,144
Number of internal edges	104,065	71,422	30,326	20,080
Number of boundary edges	223	415	129	64
Number of holes	5	1	0	1
Mesh Type	Irregular	Irregular	Irregular	Semi-regular



(a) Bunny (b) Davidhead (c) Hand (d) Head

Figure 4.7: Test models

The quality of the geometry of 3-D mesh model is measured by the forward and backward RMS (Root Mean Square) errors (Eq. (2.15)), respectively. We list the maximum value between the two RMS distance, denoted as E_{mrms}. To measure the robustness, the correlation coefficient between extracted watermark bits $\omega_n^{''}$ and the designated watermark ω_n is calculated by means of

$$corr(\omega_n, \omega_n^{''}) = \frac{\sum_{n=0}^{N-1} (\omega_n^{''} - \varpi^{''})(\omega_n - \varpi)}{\sqrt{\sum_{n=0}^{N-1} (\omega_n^{''} - \varpi^{''})^2 \times \sum_{n=0}^{N-1} (\omega_n - \varpi)^2}} \tag{4.5}$$

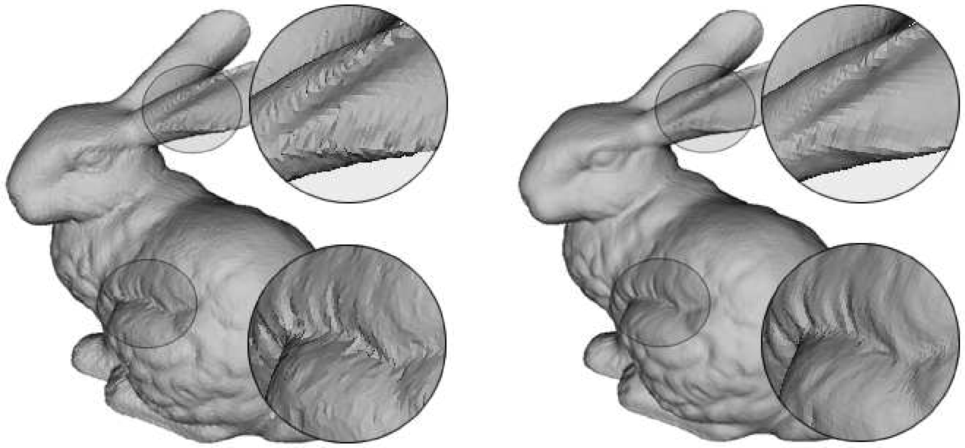
where ϖ indicates the average of the watermark and $corr$ exists in the range of $[-1, 1]$.

Table 4.2: Evaluation of invisibility, E_{mrms} of the watermark in the case of no attack. Here, α denotes the strength factor and N is the number of embedded watermark bits

Level	α, N	*Bunny*	*Davidhead*	*Hand*	*Head*
$J-1$	$0.3, 125$	0.21×10^{-4}	86.57×10^{-4}	1.85×10^{-4}	614.36×10^{-4}
$J-1$	$0.4, 125$	0.28×10^{-4}	115.63×10^{-4}	2.47×10^{-4}	819.89×10^{-4}
$J-1$	$0.4, 265$	0.28×10^{-4}	114.71×10^{-4}	2.46×10^{-4}	826.01×10^{-4}
$J-2$	$0.2, 50$	0.28×10^{-4}	92.92×10^{-4}	2.86×10^{-4}	1552.10×10^{-4}
Multilevel[a]	–	0.40×10^{-4}	147.34×10^{-4}	3.76×10^{-4}	1749.07×10^{-4}

[a] The watermark is embedded into two resolution levels at the same time. Here, the strength factor $\alpha = 0.4$ at resolution level $J-1$ with the watermark capacity $N = 265$, and $\alpha = 0.2$ for $J-2$ with $N = 50$.

In order to show the efficiency of the invisibility and robustness of watermark, we embedded the watermark varying the embedding levels ($J-1$, $J-2$ or both two levels at the same time), the watermark strength factor α and the number of watermark bits N. Note that J denotes the full resolution levels, i.e., spatial domain (see Fig. 2.11). The



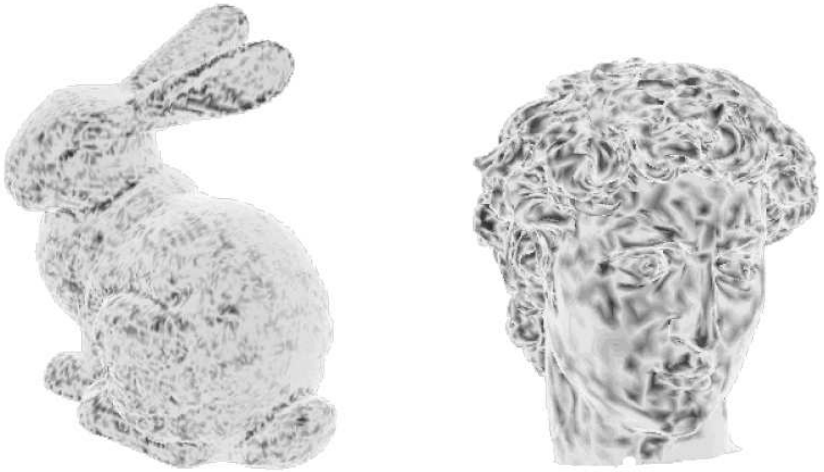
(a) $E(V, V') = 0.60 \times 10^{-4}$

(b) $E(V, V') = 0.69 \times 10^{-4}$

Figure 4.8: $E(V, V')$ for *Bunny* according to embedding level (a) $J - 1$ where the watermark strength factor $\alpha = 0.8$, and (b) $J - 4$ where $\alpha = 0.1$.

results are given in Table 4.2. In the same level, the robustness increases in proportion to the watermark strength factor α, but the visual distortion E_{mrms} also increases with α. The invisibility of the watermark according to the watermark capacity N is also given in the table. When we embedded 125 and 265 bits at resolution level $J - 1$ with fixed strength factor $\alpha = 0.4$, E_{mrms} are almost same between two models. The invisibility and robustness of the watermark is not depend on the number of watermark bits in our methods. When the E_{mrms} of resolution level $J - 2$ are compared with that of $J - 1$, the E_{mrms} increases in the case of 'Hand' and 'Head'. That is, the watermarked meshes at the lower resolution levels $(J-2)$ are more visually distorted than that at the higher frequency. However, in the case of 'Davidhead' model which has many vertices with high curvatures in the part of hair, we see the E_{mrms} decreases when the embedding level is $J - 2$. Note that the smaller E_{mrms} does not always shows the better invisibility of the watermark. We compare two similar E_{mrms} of resolution level $J - 1$ and $J - 4$ with a test model, 'Bunny'. Since the magnitude of the wavelet coefficient norms at $J - 4$ is bigger than that at $J - 1$, the watermark strength factors are properly controlled at each level. We have used $\alpha = 0.8$ at resolution level $J - 1$ and $\alpha = 0.1$ at $J - 4$ level. The watermarked meshes are shown in Fig. 4.8. The figure depicts that the watermark at the higher resolution level $J - 1$ is more visible than that at the lower resolution level $J - 4$ even though two models have similar E_{mrms}. This is the fact that the modification at the lower frequency component is less affected to high curvature area of the model. It proves that the methods based on frequency domain are more effective than spatial domain methods when the application needs a good visual quality of the model after the watermarking.

Fig. 4.9 shows two watermarked meshes, Bunnt and Davidhead from Table 4.2. The watermark is embedded into two resolution levels at the same time. Here, the strength factor $\alpha = 0.4$ at resolution level $J-1$ with the watermark capacity $N = 265$, and $\alpha = 0.2$ for $J-2$ with $N = 50$. Note that dark region indicates the distorted part caused by watermark embedding. As we have expected, the higher watermark capacity is available by embedding the watermark into the wavelet coefficients of two resolution levels at same time, but the visual distortion also increases.



(a) $E_{mrms} = 0.40 \times 10^{-4}$ (b) $E_{mrms} = 147.34 \times 10^{-4}$

Figure 4.9: Watermarked 'Bunny' and 'Davidhead' which are watermarked at two resolution levels ($J-1$ and $J-2$) at same time. Dark region indicates the distorted one caused by watermark embedding

Table 4.3: Evaluation of the robustness against additive noise attacks when embedding at $J-1$

Error rate	(α, M)	*corr*			
		Bunny	*Davidhead*	*Hand*	*Head*
0.2%	$0.3, 125$	0.952	1.000	0.984	0.890
	$0.4, 125$	1.000	1.000	1.000	0.968
	$0.4, 265$	0.940	1.000	1.000	0.903
0.45%	$0.3, 125$	0.744	1.000	1.000	0.615
	$0.4, 125$	0.856	1.000	1.000	0.744
	$0.4, 265$	0.751	1.000	0.992	0.601

As we mentioned in Section 4.2.2, the capacity and the robustness of our method is dependent on the number of wavelet coefficients. For the fair analysis of robustness, we embed the same number of bits with the same strength factor α to all test models. We give the result from single level at $J-1$ in Table 4.3. For the noise attacks, binary random noise was add to each vertex of the watermarked model with different error rates: 0.2% and

Table 4.4: Evaluation of the robustness against Laplacian smoothing attacks when embedding at $J-1$

Iterations	(α, M)	*corr*			
		Bunny	*Davidhead*	*Hand*	*Head*
20	$0.3, 125$	0.920	1.000	0.906	0.936
	$0.4, 125$	1.000	1.000	1.000	1.000
	$0.4, 265$	0.947	1.000	0.970	0.962
40	$0.3, 125$	0.795	0.968	0.744	0.588
	$0.4, 125$	0.888	1.000	0.808	0.728
	$0.4, 265$	0.706	0.917	0.753	0.751

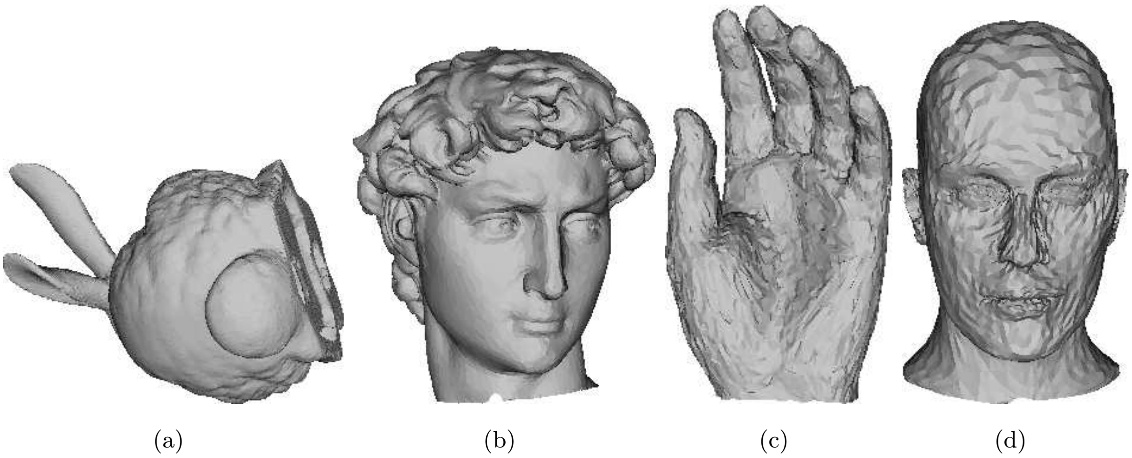


Figure 4.10: Attacked models: (a) Affine transform, (b) Laplacian smoothing (0.03 × 40), (c) uniform vertex coordinates quantization (8 bits), (d) Additive noise (0.45%).

0.45%. Here, the error rate represents the amplitude of noise as a fraction of the maximum vertex norm of the object. In Table 4.4, we apply Laplacian smoothing [FIEL88] with the relaxation factor of 0.03 and the iteration 20 and 40 times, respectively. The correlation coefficient increases according to α. But, if we increase the watermark strength factor α, it makes the watermark more visible. A small number of embedded bits gives better robustness. That is, there is a trade off between the capacity, invisibility and the robustness of the watermark in our method.

Table 4.5 presents the result of multilevel embedding. We embedded 315 bits, (265 bits at level $J-1$ and 50 bits at level $J-2$), into all test models. The strength factors are $\alpha = 0.4$ at $J-1$ and $\alpha = 0.2$ at $J-2$, respectively. The table gives the correlation coefficient at each level and the average of both levels. To evaluate the robustness against lossy compression, we apply uniform quantization of coordinates. The numbers of quantization steps are 2^8 (8 bits) and 2^9 (9 bits), and a small number of bits gives the lower quality of the model.

Table 4.5: Robustness against additive noise ($AN^{amplitude}$) and Laplacian smoothing ($LS^{numbers\ of\ iteration}$) and uniform quantization (UQ^{bits}), random vertex re-ordering (RR), similarity transform (ST), non-uniform scaling ($NS^{(x,\ y,\ z)}$), affine transform (RST) of *multilevel* embedding.

	Level	*Bunny*		*DavidHead*		*Hand*		*Head*	
		corr	*Avg.*	*corr*	*Avg.*	*corr*	*Avg.*	*corr*	*Avg.*
$AN^{0.2\%}$	$J-1$	0.940	0.970	1.000	1.000	1.000	1.000	0.903	0.951
	$J-2$	1.000		1.000		1.000		1.000	
$AN^{0.45\%}$	$J-1$	0.751	0.876	1.000	1.000	1.000	1.000	0.601	0.801
	$J-2$	1.000		1.000		1.000		1.000	
LS^{20}	$J-1$	0.955	0.977	1.000	1.000	0.962	0.902	0.962	0.981
	$J-2$	1.000		1.000		0.842		1.000	
LS^{40}	$J-1$	0.698	0.809	0.917	0.900	0.723	0.666	0.766	0.863
	$J-2$	0.919		0.883		0.610		0.960	
UQ^{8}	$J-1$	0.992	0.996	1.000	1.000	0.992	0.996	0.584	0.792
	$J-2$	1.000		1.000		1.000		1.000	
UQ^{9}	$J-1$	0.947	0.974	1.000	1.000	1.000	1.000	0.940	0.970
	$J-2$	1.000		1.000		1.000		1.000	
RR	$J-1$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	$J-2$	1.000		1.000		1.000		1.000	
ST	$J-1$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	$J-2$	1.000		1.000		1.000		1.000	
$NS^{(2.0,1.0,1.0)}$	$J-1$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	$J-2$	1.000		1.000		1.000		1.000	
$NS^{(1.0,2.0,1.0)}$	$J-1$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	$J-2$	1.000		1.000		1.000		1.000	
$NS^{(1.0,1.0,2.0)}$	$J-1$	0.992	0.996	0.992	0.996	1.000	0.980	1.000	0.980
	$J-2$	1.000		1.000		0.960		0.960	
RST[a]	$J-1$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	$J-2$	1.000		1.000		1.000		1.000	
RST[b]	$J-1$	0.992	0.996	0.992	0.996	1.000	1.000	1.000	1.000
	$J-2$	1.000		1.000		1.000		1.000	

[a] $R^{(30°,60°,90°)}, S^{(2.0,1.1,1.7)}, T^{(5,10,15)}$

[b] $R^{(90°,30°,60°)}, S^{(1.1,1.7,2.0)}, T^{(-5,10,-15)}$

R : Rotation (ρ, θ, ϕ)

S : Scaling (a_x, a_y, a_z), where a_u is scaling factor for axis $u\,(u : x, y, z)$

T : Translation (t_x, t_y, t_z)

The resolution level $J-2$ shows the better robustness than $J-1$ because the wavelet coefficients D^{J-2} are less affected than D^{J-1} to the other attacks.

We also test similarity transform, affine transform, and random vertex re-ordering of both faces and vertex indices. Since the wavelet coefficient norms are invariant to rotation and translation, our method that embeds into wavelet coefficient norms gives error-free detection on rotation and translation for both single level and multilevel. In addition, it is also effective on scaling attacks including non-uniform scaling. Random re-ordering attack can be recovered by the re-ordering before forward wavelet transform.

4.4 Conclusions

In this chapter, a watermarking method for 3-D surface meshes was proposed. Multiresolution wavelet analysis is used to achieve both the robustness and the invisibility of the watermark. Since irregular wavelet analysis can be applied to both regular and irregular meshes, our method do not use a pre-processing such as re-meshing for the watermark embedding and extraction. To recover the topological information after inverse wavelet transform, vertex re-ordering algorithm was used as a pre-processing for forward wavelet transform and a post-processing for inverse wavelet transform. The watermarks which are embedded into the wavelet coefficients are visually imperceptible while having the good robustness against geometrical attacks and distortion-less attacks. Through the simulation results, we proved that the vertex and face re-ordering procedure as pre-processing for both watermark embedding and extraction makes our method be robust against connectivity re-ordering attacks. We also showed that multiresolution embedding has more capacity while keeping the robustness against connectivity re-ordering as well as geometrical attacks. However, the watermark which is embedded into the wavelet coefficients in a certain order can not be retrieved after topological attacks. For the robustness against topological attacks, pre-processing such as registration and re-sampling is required as in [PRAU99, YIN01], or the robustness against topological attacks cannot be guaranteed [KANA98, UCCH04, KIM05].

Chapter 5

Digital Watermarking Methods Using the Distribution of the Scale Coefficients

In this chapter, we introduce a blind watermarking method for 3-D irregular surface meshes that embeds the watermark into the approximation meshes (scale coefficients of the wavelet transform). This work has published in [KIM06a, KIM06b].

5.1 Introduction

It is very important in general to determine a watermark carrier, also called primitive, that can effectively preserve watermark from attacks. For example, if the wavelet coefficients or scale coefficients arranged in a certain order are used as the watermark carrier, the hidden watermark can not be retrieved without referring to the original model after topological attacks. This is caused by the fact that 3D polygonal meshes do not have implicit order and connectivity of vertices. For the same reason, pre-processing such as registration and re-sampling is required as in [PRAU99, YIN01], or the robustness against topological attacks cannot be guaranteed [KANA98, UCCH04, KIM05]. In this chapter, we use statistical features of scale coefficients on approximated meshes as a watermark carrier. The statistical features might be less sensitive than the scale (or wavelet) coefficients themselves in a certain order. The proposed methods modify the distribution of scale coefficients to embed the watermark. It is not necessary to use any pre-processing such as registration, re-sampling and re-ordering for wavelet analysis in the process of watermark detection. The hidden watermark is extracted, not only in the same resolution level as used in embedding process, but also in the spatial domain. Furthermore, the watermark extraction in the spatial domain can reduce the computational complexity and processing time caused by wavelet transform. While these techniques may slightly decrease the robustness against geometrical attacks comparing with a spatial domain based methods [CHO07], it allows our methods to be robust against topological attacks. As the number of scale coefficients decreases after wavelet analysis, our methods are more applicable to relatively large models.

5.2 Blind Watermarking Using the Distribution of the Scale Coefficients

For the sake of the robustness against various topological attacks, we introduce two methods that have not been employed in the previous wavelet analysis based methods: the mean modification and the variance shifting method.

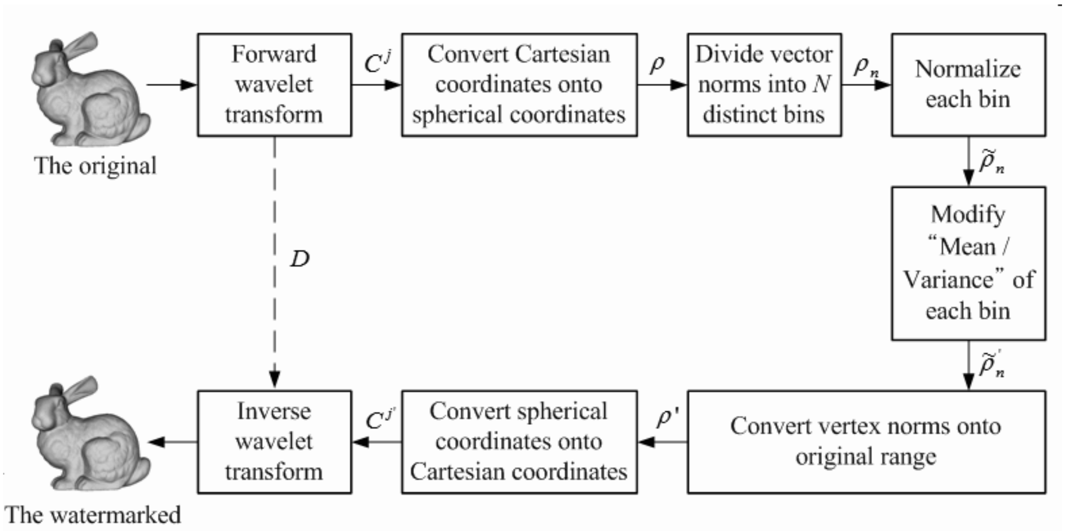

The original
Forward wavelet transform
C^j
Convert Cartesian coordinates onto spherical coordinates
ρ
Divide vector norms into N distinct bins
ρ_n
Normalize each bin
$\tilde{\rho}_n$
Modify "Mean / Variance" of each bin
$\tilde{\rho}'_n$
D
Convert vertex norms onto original range
ρ'
Convert spherical coordinates onto Cartesian coordinates
$C^{j'}$
Inverse wavelet transform
The watermarked

Figure 5.1: A block diagram of watermark embedding procedures

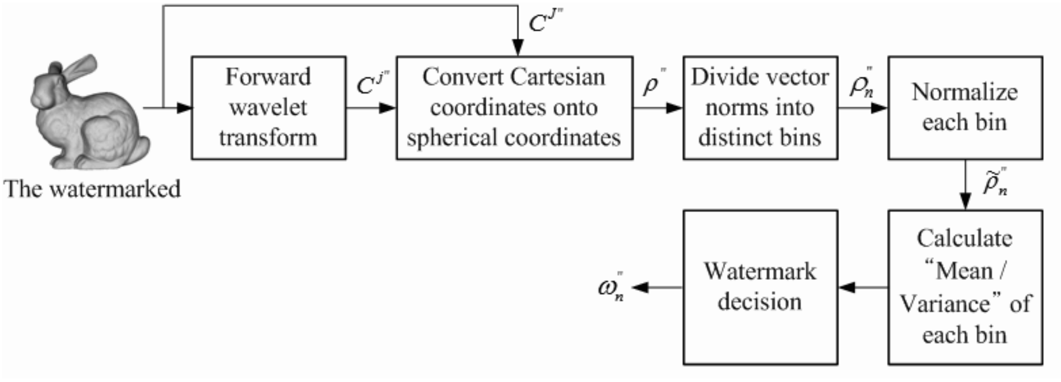
Fig. 5.1 shows the proposed watermark embedding procedures. To embed the watermark, we first perform forward wavelet analysis with the original meshes. Then, we obtain a set of the scale coefficients vector C^j at resolution level j, from Eq. (2.20) as follows:

$$C^j = [\ c_0,\ c_1, \dots\ ,\ c_i,\ c_{|C^j|-1}\ ]^T \tag{5.1}$$

where $c_i = (x_i\ y_i\ z_i)^T$, and $|C^j|$ is the number of scale coefficients. The resolution level j can be determined by considering the capacity and the invisibility of the watermark embedding. Each scale coefficient in Cartesian coordinates, c_i, is converted to spherical coordinates by means of

$$
\begin{aligned}
\rho_i &= \sqrt{(x_i - x_g)^2 + (y_i - y_g)^2 + (z_i - z_g)^2} \\
\theta_i &= tan^{-1} \frac{y_i - y_g}{x_i - x_g} \\
\phi_i &= cos^{-1} \frac{z_i - z_g}{\sqrt{(x_i - x_i)^2 + (y_i - y_g)^2 + (z_i - z_g)^2}}
\end{aligned}
\tag{5.2}
$$

where $0 \le i < |C^j|$ and (x_g, y_g, z_g) is the center of gravity of C^j. The proposed method uses only the scale coefficient norms, ρ_i, for watermarking and keeps the other two components, θ_i and ϕ_i, intact. Then, the probability distribution of ρ_i is divided into N distinct bins with equal range according to their magnitude. Each bin is used independently to hide one bit of watermark. Scale coefficients norms belonging to the n-th bin are mapped into the normalized range. The r-th normalized vertex norm in n-th bin is denoted as $\widetilde{\rho}_{n,r}$. Here, we assume that the scale coefficient norms have a uniform distribution in the normalized range. According to the watermark ($\omega_n \in \{+1, -1\}$) to be embedded, we change the mean (or the variance) value of each bin via transforming $\widetilde{\rho}_{n,r}$ by histogram mapping function [CHO07]. The normalization and distribution modification steps are different in the mean modification and the variance shifting method, respectively. All transformed norms in each bin are mapped onto the original range. Then, all the bins are combined and converted to Cartesian coordinates. Finally, inverse wavelet analysis is performed to get the watermarked meshes.



C^{J''}
The watermarked
Forward wavelet transform
C^{j''}
Convert Cartesian coordinates onto spherical coordinates
ρ''
Divide vector norms into distinct bins
ρ''_n
Normalize each bin
ρ̃''_n
Calculate "Mean / Variance" of each bin
Watermark decision
ω''_n

Figure 5.2: A block diagram of watermark extraction procedures

Fig. 5.2 shows the proposed watermark extraction procedures. The watermark detection does not require the original meshes. Watermark extraction procedures begin with forward wavelet transform followed by spherical coordinates conversion of vertex on the watermarked meshes. Note that the vertex norms are taken from the spatial domain without wavelet analysis (see Section 5.3). Similar to the watermark embedding process, the scale coefficient (or vertex) norms are classified into N bins and mapped onto the normalized range. The mean (or the variance) value of scale coefficient (for the watermark extraction in wavelet domain) or vertex norms (for the watermark extraction in spatial domain) are calculated in each bin and compared to each reference value.

5.2.1 Mean Modification Method

This method embeds the watermark information by shifting the mean value of each bin according to an assigned watermark bit. In this subsection, we describe mainly the normalization and the mean modification steps in the watermark embedding procedure and watermark decision step in the extraction procedure.

In the normalization step, the scale coefficient norms, $\rho_{n,r}$, belonging to the n-th bin are mapped into the normalized range of $[0, 1]$ by

$$\widetilde{\rho}_{n,r} = \frac{\rho_{n,r} - \min_{\rho_{n,r}}}{\max_{\rho_{n,r}} - \min_{\rho_{n,r}}} \tag{5.3}$$

where $\widetilde{\rho}_{n,r}$ is the r-th normalized norm of the n-th bin. $\max_{\rho_{n,r}}$ is the maximum norm of the n-th bin and $\min_{\rho_{n,r}}$ is the minimum norm. We assume that each bin has a distribution very close to uniform over the unit interval.

In the mean modification step, the mean of n-th bin, μ_n, is shifted by a factor $+\alpha$ (or $-\alpha$) to embed n-th watermark bit $\omega_n = +1$ (or $\omega_n = -1$), as follows:

$$\mu_n^{\prime} = \begin{cases} \frac{1}{2} + \alpha \text{ if } \omega_n = +1 \\ \frac{1}{2} - \alpha \text{ if } \omega_n = -1 \end{cases} \tag{5.4}$$

where α $(0 < \alpha < \frac{1}{2})$ is the strength factor that can control the robustness, the transparency of watermark, and $mu_n^{\prime}$ is the modified mean value. To shift the mean to the desired level, the normalized scale coefficient norms, $\widetilde{\rho}_{n,r}$, are transformed iteratively by a histogram mapping function as follows:

$$\widetilde{\rho'}_{n,r} = (\widetilde{\rho}_{n,r})^{k_n} \text{ for } 0 < k_n < \infty \text{ and } k_n \in \mathbb{R} \tag{5.5}$$

That is, the mean value decreases with relatively small value for increasing k_n on the range $]1, \infty[$. On the other hand, the mean value increases for decreasing k_n on the range $]0, 1[$.

The exact parameter k_n can be found directly from Eq. (3.8).

$$k_n = \begin{cases} \frac{1-2\alpha}{1+2\alpha} \text{ if } \omega_n = +1 \\ \frac{1+2\alpha}{1-2\alpha} \text{ if } \omega_n = -1 \end{cases} \tag{5.6}$$

Note that k_n is in the range of $]0, 1[$ when the watermark bit is $+1$, and k_n does in the range of $]1, \infty[$ when watermark bit is -1.

The real vertex norm distribution in each bin is neither continuous nor uniform. Then the parameter k_n cannot be directly calculated by Eq. (5.6). To overcome this difficulty, we use an iterative approach as follow.

For embedding $\omega_n = +1$ into the n-th bin:

1) Initialize the parameter k_n as 1;
2) Transform normalized norms by $\widetilde{\rho}'_{n,r} = (\widetilde{\rho}_{n,r})^{k_n}$;
3) Calculate mean of transformed norms through $\widetilde{\mu}'_n = \frac{1}{M_n} \sum_{m=0}^{M_n-1} \widetilde{\rho}'_{n,r}$;
4) If $\widetilde{\mu}'_n < \frac{1}{2} + \alpha$, decrease k_n ($k_n = k_n - \Delta k$) and go back to 2);
5) Replace normalized norms with transformed norms using $\widetilde{\rho}_{n,r} = \widetilde{\rho}'_{n,r}$;
6) End.

For embedding $\omega_n = -1$ into the n-th bin:

4) If $\widetilde{\mu}'_n > \frac{1}{2} - \alpha$, increase k_n ($k_n = k_n + \Delta k$) and go back to 2);

Here, Δk indicates the step size for decrement or increment of parameter k_n. Note that only the step 4) is different according to the watermark bit to be embedded.

Transformed norms of each bin are mapped onto the original range. Finally, the watermark embedding process is completed by combining all of the bins and converting the spherical coordinates to Cartesian coordinates. Let $\widetilde{\rho}'_i$ be the scale coefficient norms in the combined bin. The watermarked scale coefficients $c'_i = (x'_i, y'_i, z'_i)$ represented in Cartesian coordinate is obtained by

$$\begin{aligned} x'_i &= \rho'_i \cos\theta_i \sin\phi_i + x_g \\ y'_i &= \rho'_i \sin\theta_i \sin\phi_i + y_g \\ z'_i &= \rho'_i \cos\phi_i + z_g \end{aligned} \tag{5.7}$$

The extraction procedure is quite simple (see Fig. 5.2). The scale coefficient (or vertex) norms taken from the watermarked mesh model with (or without) wavelet analysis and are classified into N bins and mapped onto the normalized range of $[0, 1]$. The mean of each bin, μ'_n, is calculated and compared to the reference mean value, $\frac{1}{2}$. The watermark hidden in the n-th bin, ω'_n, is extracted by means of

$$\omega'_n = \begin{cases} +1, & \text{if } \mu'_n > \frac{1}{2} \\ -1, & \text{if } \mu'_n < \frac{1}{2} \end{cases} \tag{5.8}$$

5.2.2 Variance Shifting Method

This method is to change the variance value of each bin to be greater or smaller than a reference according to an assigned watermark bit. Both the watermark embedding and extraction of the method are quite similar to the previous method. The normalization step and the histogram mapping function to modify the variance are only different. More details are described in this subsections. For simplicity, we use the same notation as the mean modification method.

In the normalization step, the scale coefficient norms of each bin are mapped into the range of [-1,1] similar to the previous method :

$$\widetilde{\rho}_{n,r} = 2 \cdot \frac{\rho_{n,r} - \min_{\rho_{n,r}}}{\max_{\rho_{n,r}} - \min_{\rho_{n,r}}} - 1 \tag{5.9}$$

where $\widetilde{\rho}_{n,r}$ is the r-th norms of the n-th bin represented in the normalized range. Note that each bin has a nearly uniform distribution over the interval $[-1, 1]$.

In the variance modification step, the variance of the n-th bin, σ_n^2, is modified by a factor $+\alpha$ (or $-\alpha$) to embed the n-th watermark bit $\omega_n = +1$ (or $\omega_n = -1$), as follows:

$$\sigma_n^{2'} = \begin{cases} \frac{1}{3} + \alpha \text{ if } \omega_n = +1 \\ \frac{1}{3} - \alpha \text{ if } \omega_n = -1 \end{cases} \tag{5.10}$$

where α $(0 < \alpha < \frac{1}{3})$ is the strength factor. To modify the variance to the desired level, the normalized norms, $\widetilde{\rho}_{n,r}$, are transformed iteratively by a histogram mapping function as given by,

$$\widetilde{\rho}'_{n,r} = sign(\widetilde{\rho}_{n,r}) \, |\widetilde{\rho}_{n,r}|^{k_n} \text{ for } 0 < k_n < \infty \text{ and } k_n \in \mathbb{R} \tag{5.11}$$

where $sign(x)$ denotes the sign of x. That is, the variance value decreases with relatively small value for increasing k_n on the range $]1, \infty[$. On the other hand, the variance increases for decreasing k_n on the range $]0, 1[$.

The exact parameter can be directly found from Eq. (3.12).

$$k_n = \begin{cases} \frac{1-3\alpha}{1+3\alpha} \text{ if } \omega_n = +1 \\ \frac{1+3\alpha}{1-3\alpha} \text{ if } \omega_n = -1 \end{cases} \tag{5.12}$$

Note that k_n is in the range of $]0, 1[$ when the watermark bit is $+1$, and k_n does in the range of $]1, \infty[$ when watermark bit is -1.

As mentioned in the mean modification method, Eq. (5.12) is not useful for the real distribution of scale coefficient norms. Thus we use an iterative approach as follow.

For embedding $\omega_n = +1$ into the n-th bin:

```
1) Initialize the parameter $k_n$ as 1;
2) Transform normalized norms by $\widetilde{\rho}'_{n,r} = sign\left(\widetilde{\rho}_{n,r}\right) |\widetilde{\rho}_{n,r}|^{k_n}$;
3) Calculate the variance of transformed norms through $\widetilde{\sigma}_n^{2'} = \frac{1}{M_n} \sum_{m=0}^{M_n - 1} \widetilde{\rho}'^{2}_{n,r}$;
4) If $\widetilde{\sigma}_n^{2'} < \frac{1}{3} + \alpha$, decrease $k_n$ $(k_n = k_n - \Delta k)$ and go back to 2);
5) Replace normalized norms with transformed norms using $\widetilde{\rho}_{n,r} = \widetilde{\rho}'_{n,r}$;
6) End.
```

For embedding $\omega_n = -1$ into the n-th bin:

4) If $\widetilde{\sigma}_n^{2'} > \frac{1}{3} - \alpha$, increase k_n ($k_n = k_n + \Delta k$) and go back to 2);

Watermark extraction process for this method is also quite simple. The variance of each bin, $\sigma_n^{2'}$, is calculated and compared with the threshold value, $\frac{1}{3}$. The watermark hidden in the n-th bin, ω'_n, is extracted by means of

$$\omega'_n = \begin{cases} +1, & \text{if } \sigma_n^{2'} > \frac{1}{3} \\ -1, & \text{if } \sigma_n^{2'} < \frac{1}{3} \end{cases} \tag{5.13}$$

5.3 Experimental Results

We show the experimental results in this section. We two test models: 'Stanford bunny' (with 34,834 vertices and 69,451 faces), davidhead (with 24,085 vertices and 47,753 faces). The quality of the geometry of 3-D mesh model is measured by the maximum RMS (Root Mean Square) distance by Eq. (2.15) and denoted as E_{mrms}. The robustness is evaluated by correlation coefficient (*corr*), between the designed and extracted watermark by Eq. (4.5).

To evaluate the performance of the proposed methods, 64 bits of the watermark were embedded into the scale coefficients (approximation mesh) at the $(J-1)$-th resolution level C^{J-1}, as example. Table 5.1 shows the performances in terms of E_{mrms} and *corr* in the case of no attack. Here, the strength factors of each watermark are also listed. The strength factor α is determined experimentally so that both methods have similar quality for each model in terms of E_{mrms}. The original meshes and the watermarked meshes are given in Fig. 5.3. This figure depicts that the mean modification method maintains better visual quality than the variance modification method. Some artifacts appear in smooth regions such as the skin of 'Davidhead' in Fig. 5.3-(f). This is mainly due to the fact that every scale coefficient is modified without considering local curvature of models. It is also caused by discontinuities in the boundaries of neighbor bins when the distribution is modified [CHO07]. As results, our methods are more applicable to the models with a lot of bumpy area. The watermark can be extracted from the scale coefficients at the same resolution level as used in watermark embedding, and its correlation coefficients are listed as $corr^{J-1}$. Note that our method can also extract the watermark in spatial domain, since the hidden watermark of a specific resolution level is spread and therefore exists in spatial domain. We also list its results in next column (*corr* of C^{J-1}). As a reference, the experimental results with the spatial domain (the full resolution level C^J) based statistical methods [CHO07] are given in the table. As expected, wavelet analysis based methods with detecting the watermark at the $(J-1)$-th level have similar watermark detection rate to the spatial domain based methods (*corr* of C^J), but our proposed methods with detecting at the J-th level shows slightly lower detection rate. This is mainly caused by the fact that wavelet coefficients at the $(J-1)$-th level act like noise when the hidden watermark is extracted in the spatial domain (C^J).

Table 5.1: Evaluation of watermarked meshes in the case of no attack

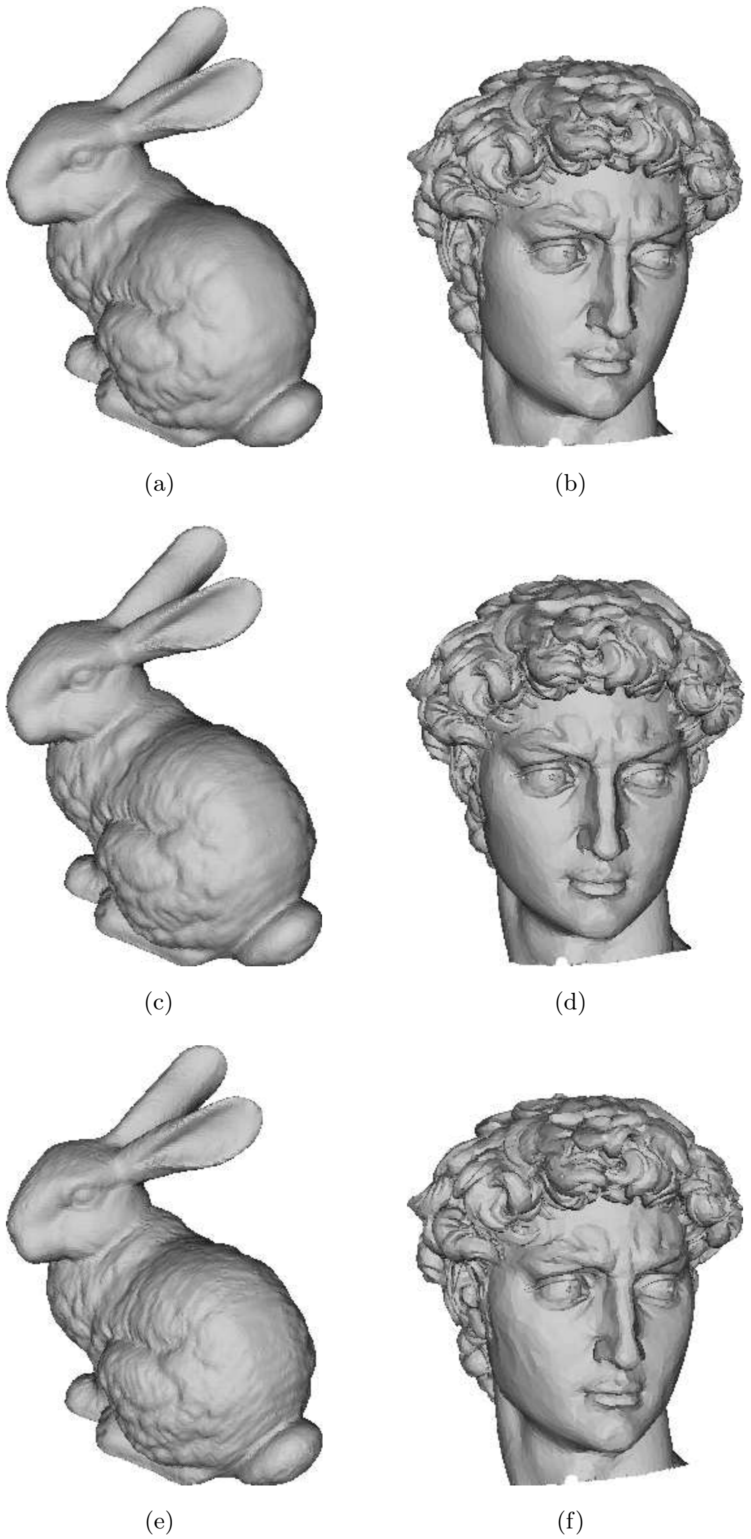
Method	Model	α	E_{mrms}	C^{J-1}		C^J
				$corr^{J-1}$	$corr$	$corr$
Mean	bunny	0.07	0.69×10^{-4}	1.00	0.88	1.00
	davidhead	0.10	137.09×10^{-4}	1.00	1.00	1.00
Variance	bunny	0.14	0.70×10^{-4}	1.00	0.94	1.00
	davidhead	0.20	136.81×10^{-4}	1.00	1.00	1.00

To evaluate the transparency of our proposed methods, we embedded the watermark into different resolution levels while keeping similar E_{mrms}. The same 64 bits of watermark information are embedded into the $(J-2)$-th (C^{J-2}), the $(J-1)$-th (C^{J-1}) and the J-th (C^J) approximation levels, respectively. The watermarked bunny models are shown in Fig. 5.3. Even though all meshes have similar $E_{mrms} \approx 0.70 \times 10^{-4}$, embedding at lower resolution level (here, C^{J-2} and C^{J-1}) is visually better than at full resolution level (C^J).

Table 5.2: Evaluation of robustness against additive binary noise attacks

Method	Model	Error rate	E_{mrms}	C^{J-1}		C^J
				$corr^{J-1}$	$corr$	$corr$
Mean	bunny	0.1%	0.97×10^{-4}	0.94	0.75	1.00
		0.3%	2.07×10^{-4}	0.53	0.53	0.53
		0.5%	3.24×10^{-4}	0.09	0.00	0.28
	davidhead	0.1%	164.85×10^{-4}	1.00	0.84	0.97
		0.3%	308.76×10^{-4}	0.38	0.25	0.38
		0.5%	472.93×10^{-4}	0.06	0.09	0.09
Variance	bunny	0.1%	0.95×10^{-4}	1.00	0.91	1.00
		0.3%	2.05×10^{-4}	-0.03	0.13	-0.09
		0.5%	3.24×10^{-4}	0.00	0.06	0.03
	davidhead	0.1%	165.69×10^{-4}	1.00	0.97	1.00
		0.3%	309.83×10^{-4}	-0.13	0.06	-0.16
		0.5%	470.70×10^{-4}	0.06	0.16	0.09

The watermarked models listed in Table 5.1 are used to evaluate the robustness against several attacks. For the noise attacks, binary random noise was added to each vertex of the watermarked model with three different error rates: 0.1%, 0.3%, 0.5%. As it is used in Chapter 4, the error rate represents the amplitude of noise as a fraction of the maximum vertex norm of the object in Table 5.2. The performance of the variance shifting method



(a) (b)

(c) (d)

(e) (f)

Figure 5.3: Test models and watermarked models, (a) Stanford Bunny, the original, (b) Davidhead, the original (c) Bunny, watermarked by the mean modification method with the strength factor $\alpha = 0.07$, and (d) Davidhead watermarked by the mean modification method with $\alpha = 0.10$, (e) bunny watermarked by the variance shifting method with $\alpha = 0.14$, and (f) Davidhead, watermarked by the variance shifting method with $\alpha = 0.20$

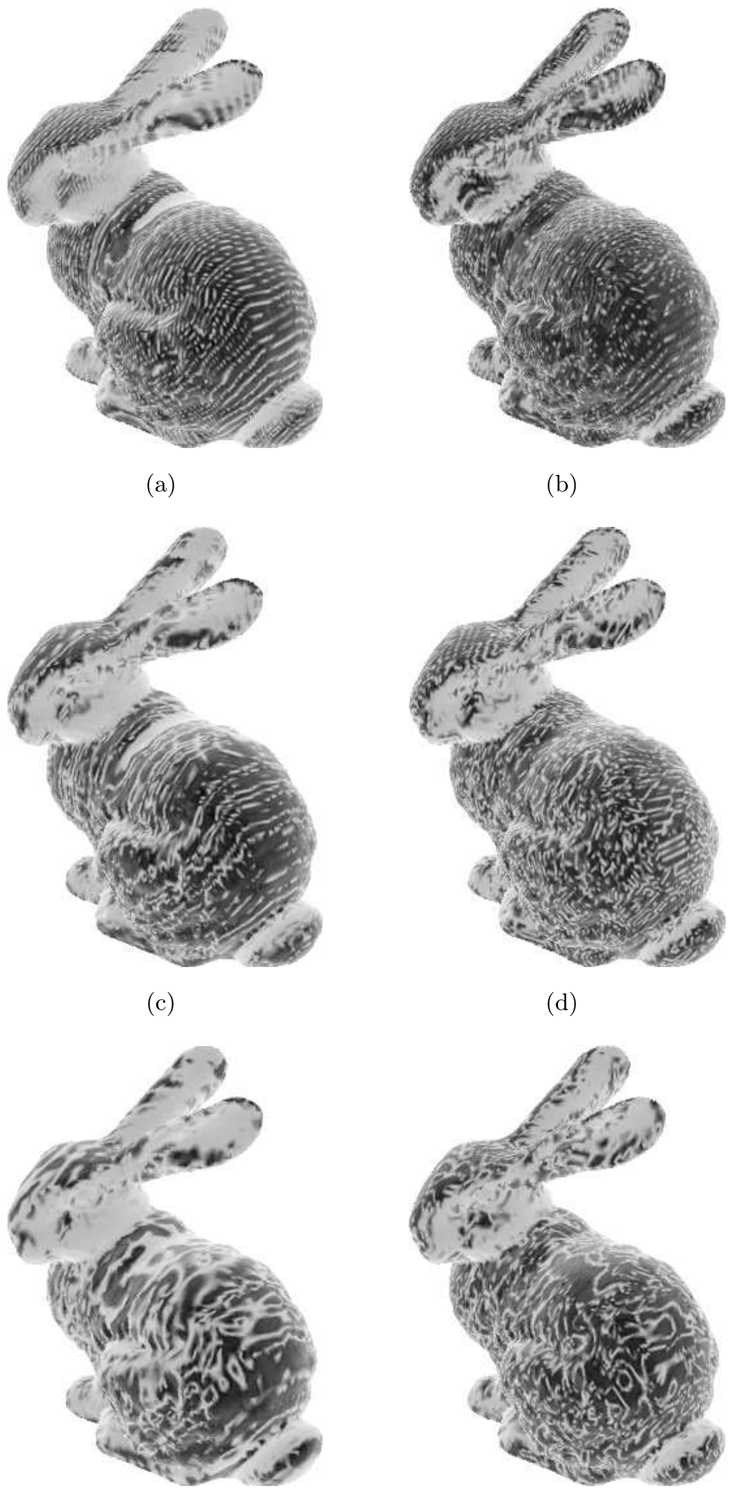


Figure 5.4: Watermark transparency when 64 watermark bits are embedded in Stanford bunny at different resolution levels, (a) C^J by the mean modification method, (b) C^J by the variance shifting method, (c) C^{J-1} by the mean modification method, (d) C^{J-1} the variance shifting method, (e) C^{J-2} by the mean modification method, (f) C^{J-2} by the variance shifting method. Dark region indicates the distorted one caused by watermark embedding.

Table 5.3: Evaluation of robustness against Laplacian smoothing attacks

Method	Model	Iteration	E_{mrms}	C^{J-1}		C^{J}
				$corr^{J-1}$	$corr$	$corr$
Mean	bunny	(10,0.03)	0.73×10^{-4}	1.00	0.85	1.00
		(30,0.03)	0.99×10^{-4}	0.88	0.75	0.87
		(50,0.03)	1.33×10^{-4}	0.76	0.70	0.84
	davidhead	(10,0.03)	179.03×10^{-4}	0.80	0.66	0.78
		(30,0.03)	352.98×10^{-4}	0.28	0.28	0.41
		(50,0.03)	515.13×10^{-4}	0.13	0.19	0.19
Variance	bunny	(10,0.03)	0.68×10^{-4}	1.00	0.91	1.00
		(30,0.03)	0.92×10^{-4}	0.94	0.91	0.91
		(50,0.03)	1.27×10^{-4}	0.88	0.75	0.06
	davidhead	(10,0.03)	178.52×10^{-4}	1.00	1.00	1.00
		(30,0.03)	354.79×10^{-4}	0.35	0.09	0.22
		(50,0.03)	518.41×10^{-4}	0.16	0.09	0.13

Table 5.4: Evaluation of robustness against uniform quantization attacks

Method	Model	Bits	E_{mrms}	C^{J-1}		C^{J}
				$corr^{J-1}$	$corr$	$corr$
Mean	bunny	9	0.92×10^{-4}	0.97	0.91	1.00
		8	1.40×10^{-4}	0.97	0.72	0.94
		7	2.53×10^{-4}	0.60	0.54	0.44
	davidhead	9	166.27×10^{-4}	1.00	0.85	1.00
		8	233.14×10^{-4}	0.72	0.66	0.91
		7	396.09×10^{-4}	0.59	0.38	0.50
Variance	bunny	9	0.91×10^{-4}	1.00	0.97	1.00
		8	1.39×10^{-4}	0.94	0.79	0.97
		7	2.51×10^{-4}	0.60	0.38	0.63
	davidhead	9	165.48×10^{-4}	0.97	1.00	1.00
		8	230.50×10^{-4}	0.94	0.91	0.91
		7	392.33×10^{-4}	0.47	0.50	0.72

decreases faster than that of the mean modification method as increasing the error rate. Both methods are fairly resistant to the noise attacks under an error rate of 0.1, but good watermark detection cannot be expected for higher error rates. This is due to the fact that the additive noise essentially alters the distribution of vertex norms in the divided bins. In addition, more vertex norms exceed the range of each bin as the noise error rate increases.

Similar tendency is observed in smoothing (Table 5.3) and uniform quantization (Table 5.4) attacks. For such reasons, the robustness cannot be enhanced beyond a certain level even when the strength factor α increases. Although the robustness can be improved by widening the size (width) of bin, the transparency of watermark and the number of bits should be carefully considered.

Table 5.3 shows the performance of the smoothing attacks [FIEL88]. Three different pairs of iteration and relaxation were applied. The robustness depends on the smoothness of the original meshes. A model that has a lot of bumpy area such as davidhead is relatively sensitive to smoothing attacks than bunny model. In Table 5.4, each vertex coordinate is quantized to 9, 8 and 7 bits for uniform quantization attacks.

The robustness against topological attacks is evaluated by simplification and subdivision. Quadric error metrics [GARL97] were used to carry out the robustness against simplification attacks. The robustness is shown in Table 5.5, where the percentage represents the number of removed vertices as a fraction of total number of vertices. It is very difficult to retrieve the hidden watermark from $(J-1)$-th resolution level in the case of simplification attacks that fatally destroy the connectivity information. This table shows that the proposed methods have fairly robustness comparable to the spatial domain based approaches (level J). In addition, the variance shifting method is more robust than the mean modification method. Subdivision attacks were also carried out. Each triangle was uniformly divided into four cells. The performance is listed in Table 5.6. The results demonstrate that the distribution of vertex norms is less sensitive to changes in the number of vertices. The proposed methods are very vulnerable to clipping attacks that causes severe alteration to the center of gravity of the model. In addition, since the watermark is embedded in wavelet domain, it cannot be perfectly extracted after vertex re-ordering. However, it can be solved by the pre- and post-processing used in Chapter 4. Clearly, our proposals can be absolutely robust against similarity transform, because the distribution of scale coefficient and vertex norms are independent to rotation, uniform scaling, and translation.

5.4 Conclusions

In this chapter, we introduced blind watermarking two methods based on the statistical characteristics of scale coefficient norms obtained by wavelet analysis for 3-D mesh model. To achieve good transparency and robustness against various topological attacks, watermark information is embedded into scale coefficients at lower resolution level by modifying their distribution and extracted, not only from the same resolution level as used in embedding process, but also directly from the spatial domain. Assuming that the distribution of each bin can be approximated uniform distribution, the mean or variance of the scale coefficients is modified by each histogram mapping function. As the watermark detect-

Table 5.5: Evaluation of robustness against simplification attacks

Method	Model	Reduction ratio	E_{mrms}	C^{J-1}		C^J
				$corr^{J-1}$	$corr$	$corr$
Mean	bunny	30%	0.75×10^{-4}	-	0.72	1.00
		50%	0.80×10^{-4}	-	0.69	0.88
		70%	0.92×10^{-4}	-	0.48	0.58
		90%	3.44×10^{-4}	-	0.06	0.42
	davidhead	30%	352.37×10^{-4}	-	0.91	0.85
		50%	422.06×10^{-4}	-	0.78	0.81
		70%	581.23×10^{-4}	-	0.85	0.85
		90%	1015.49×10^{-4}	-	0.16	0.35
Variance	bunny	30%	0.73×10^{-4}	-	0.78	1.00
		50%	0.77×10^{-4}	-	0.81	0.97
		70%	0.90×10^{-4}	-	0.81	0.94
		90%	3.12×10^{-4}	-	0.76	0.88
	davidhead	30%	509.21×10^{-4}	-	0.91	1.00
		50%	573.84×10^{-4}	-	0.91	0.94
		70%	695.56×10^{-4}	-	0.91	0.94
		90%	1419.40×10^{-4}	-	0.34	0.28

Table 5.6: Evaluation of robustness against 1-to-4 subdivision attacks

Method	Model	E_{mrms}	C^{J-1}		C^J
			$corr^{J-1}$	$corr$	$corr$
Mean	bunny	0.69×10^{-4}	-	0.76	1.00
	davidhead	136.29×10^{-4}	-	0.48	0.94
Variance	bunny	0.69×10^{-4}	-	0.88	1.00
	davidhead	135.04×10^{-4}	-	0.94	0.97

ion processes in spatial domain do not require wavelet analysis or any pre-processing, the embedded watermark can be simply calculate the mean or variance of each bin. Through simulations, we proved that the proposed methods have fairly good performance in terms of the watermark transparency and robustness against various attacks. Even though the proposed methods are not highly robust comparing with spatial domain based methods, our attempts demonstrate a possibility, blind watermarking based on wavelet analysis for 3-D mesh model.

Part II

Wavelet Transform Based Digital Watermarking for 3-D Mesh Sequences

Chapter 6

Digital Watermarking Using the Statistical Features of Distribution

In this chapter, we introduce digital watermarking methods for 3-D mesh sequences based on temporal wavelet transform. This work was published in [KIM06c].

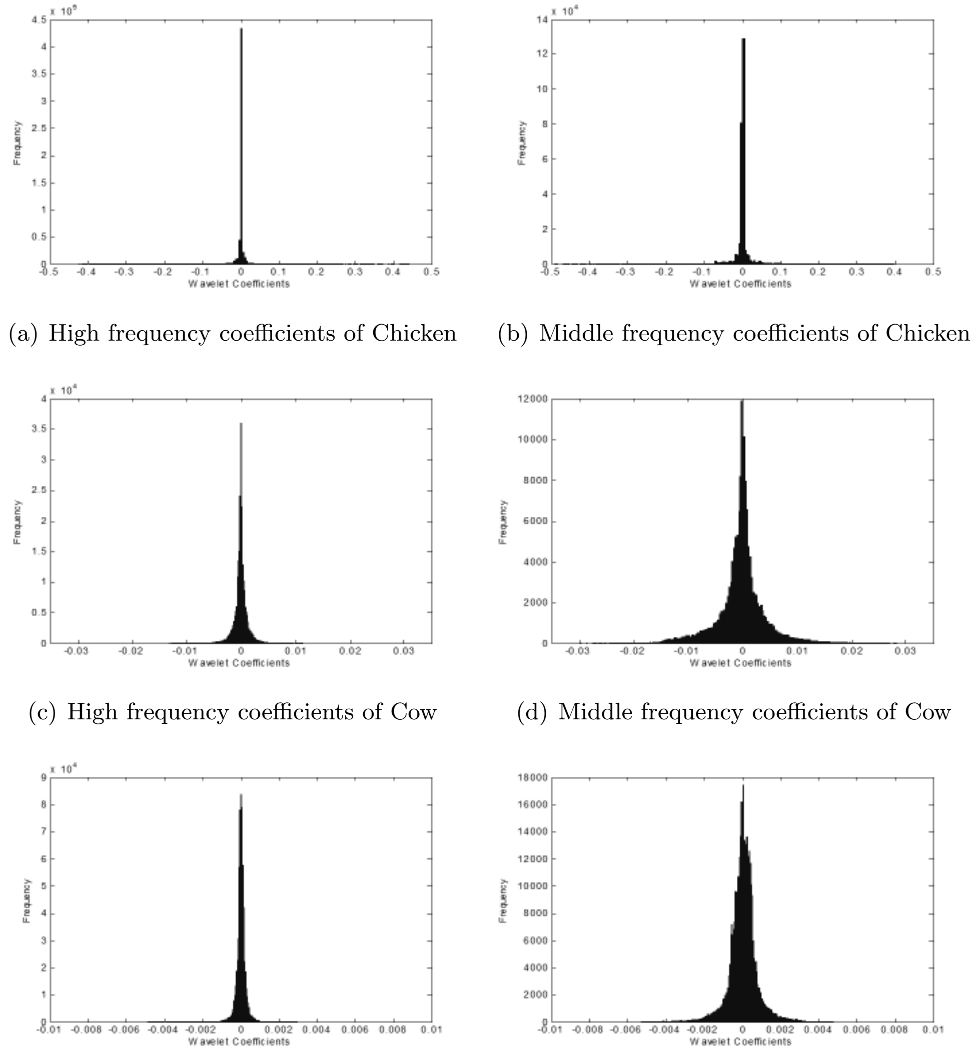
6.1 Introduction

In the previous chapter (Chapter 5), we proved that statistical features of the signal such as the mean value and the variance can be a good watermark embedding carrier. Similar to our previous works, we propose, in this chapter, two 3-D mesh sequence watermarking methods using the variance of temporal wavelet coefficients.

Assuming that the original mesh sequence is isomorphic one (i.e., the connectivity information of original 3-D mesh sequence is never changed over all frames), it can be regarded as a set of 1-D signals which consists of the geometry information with the same vertex index along the time axis.

The proposed methods decompose each 1-D signal into several sub-bands, and modify a specific sub-band to embed watermark bits. To decompose 1-D signal, wavelet transform is applied to these 1-D signals.

In order to select the sub-band for watermark embedding, we should consider the robustness and the invisibility. From the view point of the robustness, the watermark which is embedded in low frequency band can be more efficient. It is caused by the fact that the low frequency band signal is less sensitive to most signal processing since it contains most energy of signal. From the view point of the transparency, the watermark which is embedded in high frequency can be more efficient. In order to use the advantages of each sub-band, we modify the high frequency band signal to embed watermark, referring the lowest frequency band signal. Since wavelet transform can decompose both frequency components as well as temporal components, we can easily select high frequency coefficients



(a) High frequency coefficients of Chicken (b) Middle frequency coefficients of Chicken

(c) High frequency coefficients of Cow (d) Middle frequency coefficients of Cow

(e) High frequency coefficients of Dance (f) Middle frequency coefficients of Dance

Figure 6.1: Histogram of wavelet coefficients of vertex norm after temporal wavelet transform

that corresponds to low frequency coefficients. This is the reason why we use wavelet transform in order to embed watermark in this work.

For embedding a watermark, we modify the variance of wavelet coefficients in high (or middle) frequency band. The distribution of temporal wavelet coefficients can be approximated to Laplacian distribution which concentrates on zero mean as shown in Fig. 6.1. In this figure, 'Chicken', 'Cow' and 'Dance' sequences are used for practical examples. From this figure, we see that the distribution can be divided into two groups, called negative and positive group, on the basis of zero. To change the Laplacian distribution, we use an asymmetrical modification method. Fig. 6.2 shows the main idea of the proposed methods. Note that the difference between the first and the second method is just whether each axis

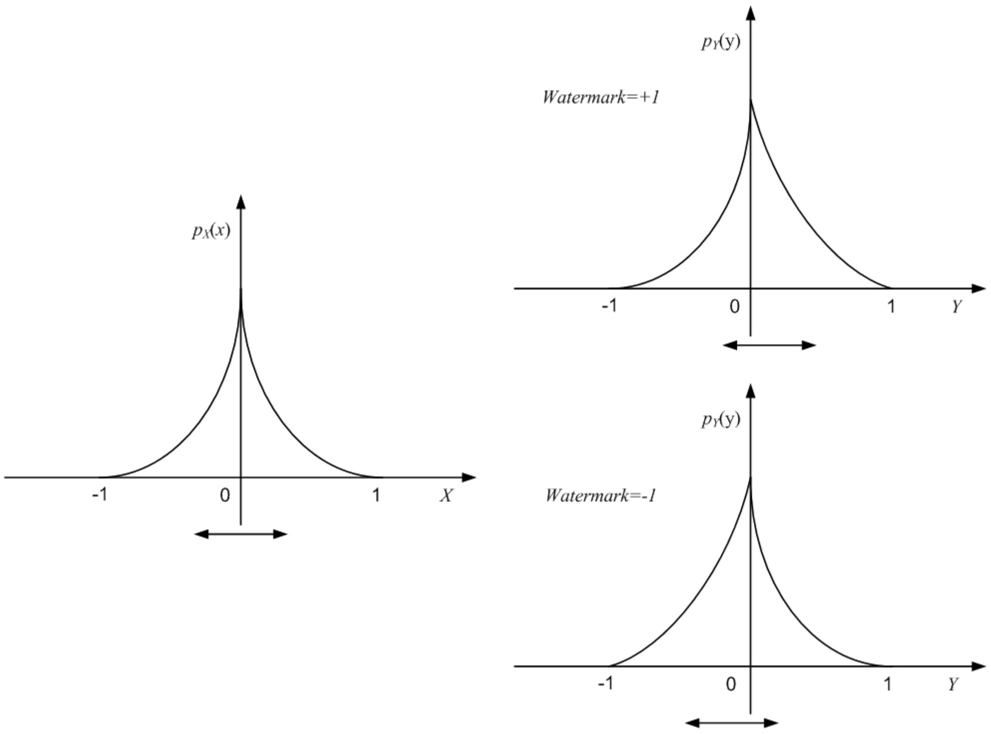


Figure 6.2: Asymmetric modification of Laplasian distribution according to the watermark bits.

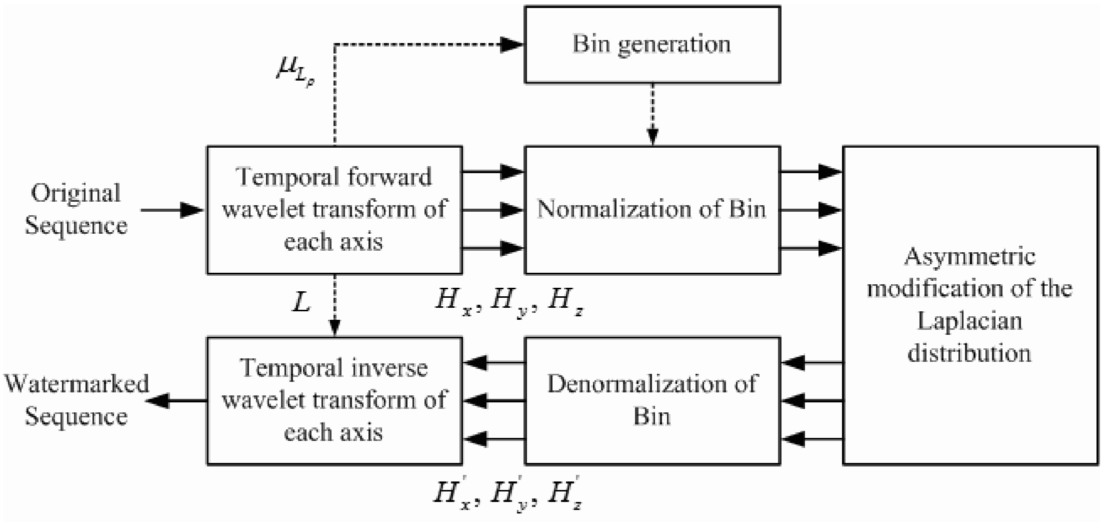
or the vector norm of 1-D signal is modified. Both of two methods can be explained by this figure. Here, we assume that the wavelet coefficients has a Laplacian distribution on the range of $[-1, 1]$. For embedding watermark $+1$, the wavelet coefficients of the positive side are modified to have bigger variance than the negative side. For embedding watermark -1, the wavelet coefficients of the negative side are modified to have bigger variance than the positive side. Here, the histogram mapping function which is used in Chapter 5 is also employed to effectively change the variance in terms of the robustness and the invisibility of watermark.

Similar to our previous methods introduced in Chapter 5, the hidden watermark can be easily extracted just by comparing the variances of one group and of the other. Note that the proposed methods are able to provide blind detection.

In the subsections 6.2 and 6.3, we respectively describe two methods in detail. The first method directly embeds the watermark into each axis of wavelet coefficients, and the second method into their vector norm. These two methods have complement performances in terms of the robustness, the invisibility and the capacity. As a sophisticated scheme, especially, the second method is combined with spatial based method [CHO07] in order to guarantee the robustness against single frame attacks. The subsection 6.4 shows the experimental results including the performance evaluations against several attacks such as frame-by-frame and frame averaging which should be considered in 3-D mesh sequence watermarking. Finally, we conclude this chapter in the subsection 6.5.

6.2 Blind Watermarking Using High Frequency Coefficients of Each axis

This proposal is to modify the variance of high (or middle) frequency coefficients of each axes, independently. Fig. 6.3 depicts the proposed watermark embedding procedures. For the sake of notational simplicity, we explain the embedding of the watermark into only an axis of the high frequency coefficients where the temporal wavelet decomposition level is one. The watermark embedding procedures begin from the original 3-D dynamic mesh sequence S ($s_t \in S, 0 \leq t < |S|$, where $|S|$ is the number of frames). Each coordinate of vertex $v_t(i) = (x_t(i), y_t(i), z_t(i))$ is transformed along the time axis using wavelet transform. The original sequence is decomposed into low frequency sequences, $l_t(i)$ $(0 \leq t < |S|/2)$, and high frequency sequences, $h_t(i)$ $(0 \leq t < |S|/2)$ [PAYA05].



μ_{L_p}
Bin generation
Original Sequence
Temporal forward wavelet transform of each axis
Normalization of Bin
Asymmetric modification of the Laplacian distribution
L
H_x, H_y, H_z
Watermarked Sequence
Temporal inverse wavelet transform of each axis
Denormalization of Bin
$H_x^{'}, H_y^{'}, H_z^{'}$

Figure 6.3: Block diagram of the watermark embedding procedures of the first method

Low frequency coefficients are used to find the vertex index in each bins as we mentioned in Section 6.1. Prior to build the bins, the average frame of low frequency sequences, $\mu_l(i)$ is calculated by

$$\mu_l(i) = \frac{1}{|S|/2} \sum_{t=0}^{(|S|/2)-1} l_t(i) \tag{6.1}$$

where $0 \leq i < |V|$ ($|V| = |V_t|$) and $0 \leq t < |S|/2$. Then, the vertex norm of the average frame is calculated by

$$\mu_{l_\rho}(i) = \sqrt{\mu_{l_{(x-x_g)^2}}(i) + \mu_{l_{(y-y_g)^2}}(i) + \mu_{l_{(z-z_g)^2}}(i)} \tag{6.2}$$

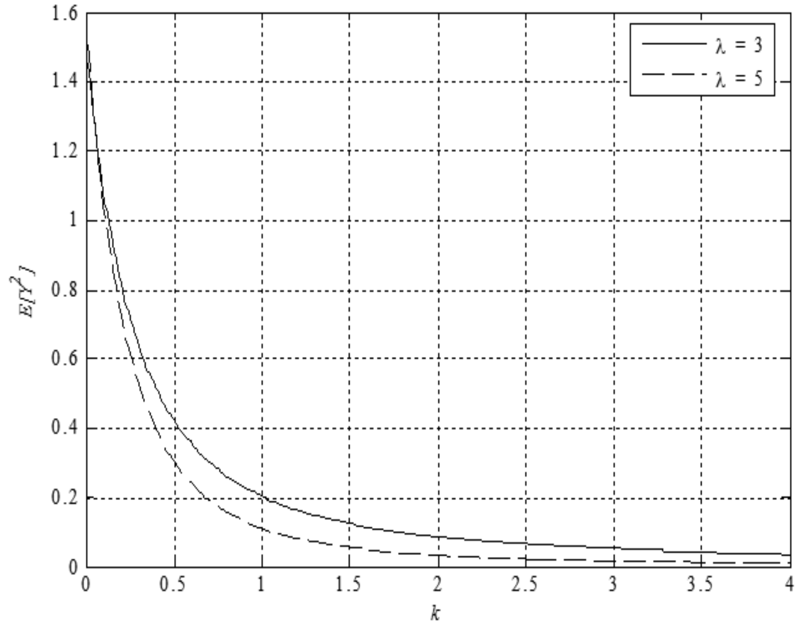


Figure 6.4: The variance of the random variable via histogram mapping function with different k, assuming that the input variable has Laplacian distribution over $[0, 1[$

where $(x_g, y_g, z_g$ is the center of gravity of the average frame and $\mu_{l_\rho}(i)$ is the vertex norm of the average frame. $\mu_{l_\rho}(i)$ is divided into N distinct bins with equal range according their magnitudes. The vertex index i in n-th bin of $\mu_{l_\rho}(i)$, will be kept to modify their corresponding high frequency coefficients, $h_{t,n}(i)$.

High frequency coefficients belonging to n-th bin, $h_{t,n}(i)$, are mapped into normalized range of $[-1, 1]$ by

$$\widetilde{h}_{t,n}(i) = \frac{h_{t,n}(i)}{max\,(h_{t,n}(i))} \tag{6.3}$$

where $\widetilde{h}_{t,n}(i)$ is the normalized high frequency coefficients. Now, we assume that the distribution of $\widetilde{h}_{t,n}(i)$ is Laplacian distribution with symmetric property in the range of $[-1, 1]$. High frequency coefficients can be divided into two groups, positive and negative side, by the sign of $\widetilde{h}_{t,n}(i)$.

Before moving to next step of the watermark embedding, we consider a continuous random variable X with Laplacian distribution over $[0, 1[$.

$$p_X(x) = \frac{\lambda}{1 - e^{-1}} \cdot e^{-\lambda x} \tag{6.4}$$

for $0 \leq x < 1$. For given X, the histogram mapping function is defined by

$$Y = X^k \tag{6.5}$$

where Y is the transformed variable, and k is a real value for $0 < k < \infty$. The variance of output random variable, $E[Y^2]$ is represented as

$$E[Y^2] = \int_0^1 x^{2k} p_X(x)dx = \frac{1}{1-e^{-1}} \sum_{n=0}^{\infty} \frac{(-1)^n \cdot \lambda^{n+1}}{n! \cdot (n+2k+1} \qquad (6.6)$$

Fig. 6.4 shows the variance of the output random variable over k of the mapping function. Note that the variance decreases monotonically as k decreases. The variance can be adjusted by selecting proper parameter. We have only shown the positive range. This can be also derived to a continuous random variable with Laplacian distribution over $[-1, 0[$.

To embed n-th watermark bit $\omega_n \in \pm 1$, the variance of the normalized high frequency coefficients in the n-th bin $\widetilde{h}_{t,n}(i)$ is modified such that

$$\begin{aligned} E[Y^2_{n,+}] &> E[Y^2_n](1+\alpha) \quad \text{if} \quad \omega_n = +1 \\ E[Y^2_{n,+}] &< E[Y^2_n](1-\alpha) \quad \text{if} \quad \omega_n = -1 \end{aligned} \qquad (6.7)$$

where $E[Y^2_{n,+}]$ denote that the variance of the positive side of the distribution of $\tilde{s}_{2t+1,n}(i)$, α is the strength factor that can control the robustness and transparency of watermark. To modify the variance to the desired level, high frequency coefficients, $\tilde{s}_{2t+1,n}(i)$, are transformed iteratively by a histogram mapping function given by

$$\widetilde{h}'_{t,n}(i) = \text{sign}(\widetilde{h}_{t,n}(i)) \cdot \left|\widetilde{h}_{t,n}(i)\right|^{k_n} \quad \text{for} \quad 0 < k_n < \infty \quad \text{and} \quad k_n \in \mathbb{R} \qquad (6.8)$$

When the parameter k_m is selected in $]1, \infty[$, $\widetilde{h}_{t,n}(i)$ are transformed into $\widetilde{h}'_{t,n}(i)$ while maintaining their signs. Moreover, the absolute value of transformed variable becomes smaller as increasing k_n. It means a reduction of the variance. On the other hands, the variance increases for decreasing k_n on the range $]0, 1[$.

All transformed high frequency coefficients in each bin are mapped onto the original range. Then, temporal inverse wavelet transform is performed to get the watermarked 3-D dynamic mesh sequence. Fig. 6.5 depicts the proposed watermark extraction procedures.

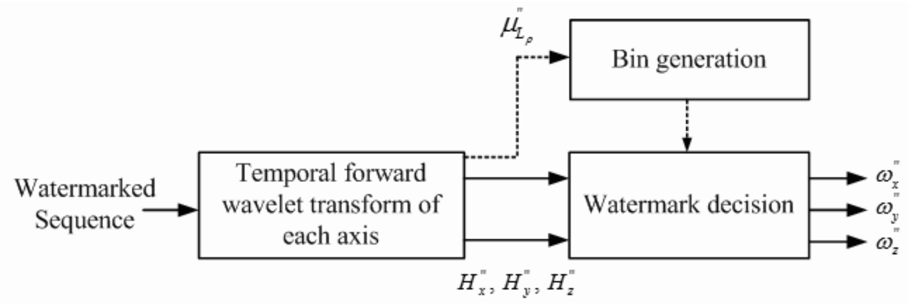


Figure 6.5: Block diagram of the watermark extraction procedures of the first method

Similar to the watermark embedding, each axis of vertex coordinates of the watermarked 3-D mesh sequence S'' is temporal wavelet transformed. The vertex norm of the average frame $\mu_{l''_\rho}(i)$ divided into N distinct bins with equal range according their magnitudes. High frequency coefficients belonging to n-th bin, $h''_{t,n}(i)$, are mapped into normalized range of $[-1,1]$. The normalized high frequency coefficients $\widetilde{h}'_{t,n}(i)$ in each bin are grouped into positive and negative side according to the sign of $\widetilde{h}'_{t,n}(i)$. The variance of positive and negative side, $E[Y^{2''}_{n,+}]$ and $E[Y^{2''}_{n,-}]$, of n-th bin, is calculated and compared each other. The watermark hidden in the n-th bin is extracted by means of,

$$\omega''_n = \begin{cases} +1, & \text{if } \; E[Y^{2''}_{n,+}] > E[Y^{2''}_{n,-}] \\ -1, & \text{if } \; E[Y^{2''}_{n,+}] < E[Y^{2''}_{n,-}] \end{cases} \tag{6.9}$$

Note that the watermark extraction process does not require the original mesh sequence.

6.3 Blind Watermarking Using High Frequency Coefficients of Vertex Norms

The proposed method in this section also uses the distribution of temporally high frequency coefficients. In sharp contrast to the first method, vertex norms are only used in the watermark embedding and extraction procedures. In addition, we propose combined watermarking method to modify the distribution of low frequency coefficients by the method introduced in [CHO07] as well as high frequency coefficients of vertex norms. For the sake of notational simplicity, we use the same notation as the previous section.

6.3.1 Modification of High Frequency Coefficients of Vertex Norms

Fig. 6.6 depicts the watermark embedding the method. From the original 3-D dynamic mesh sequence S the center of gravity of each frame is translated to the origin as pre-processing to compute wavelet coefficients. Each frame s_t, all vertices of Cartesian coordinates, $v_t(i) = (x_t(i), y_t(i), z_t(i))$, are converted to Spherical coordinates, $v_t(i) = (\rho_t(i), \theta_t(i), \rho_t(i))$ by

$$\begin{aligned} \rho_t(i) &= \sqrt{(x_t(i)-\mu_{x_t})^2 + (y_t(i)-\mu_{y_t})^2 + (z_t(i)-\mu_{z_t})^2} \\ \theta_t(i) &= \tan^{-1}\frac{y_t(i)-\mu_{y_t}}{x_t(i)-\mu_{x_t}} \qquad \text{for} \quad 0 \le i < |v| \;\; \text{and} \;\; 0 \le t < |S| \\ \phi_t(i) &= \cos^{-1}\frac{z_t(i)-\mu_{z_t}}{\sqrt{(x_t(i)-\mu_{x_t})^2 + (y_t(i)-\mu_{y_t})^2 + (z_t(i)-\mu_{z_t})^2}} \end{aligned} \tag{6.10}$$

where $(\mu_{x_t}, \mu_{y_t}, \mu_{z_t})$ is the center of gravity of s_t.

The vertex norm, $\rho_t(i)$ is transformed along the time axis using wavelet transform. The original sequence is decomposed into low frequency coefficients, $l_t(i)$ $(0 \le t < |S|/2)$, and high frequency coefficients, $h_t(i)$ $(0 \le t < |S|/2)$.

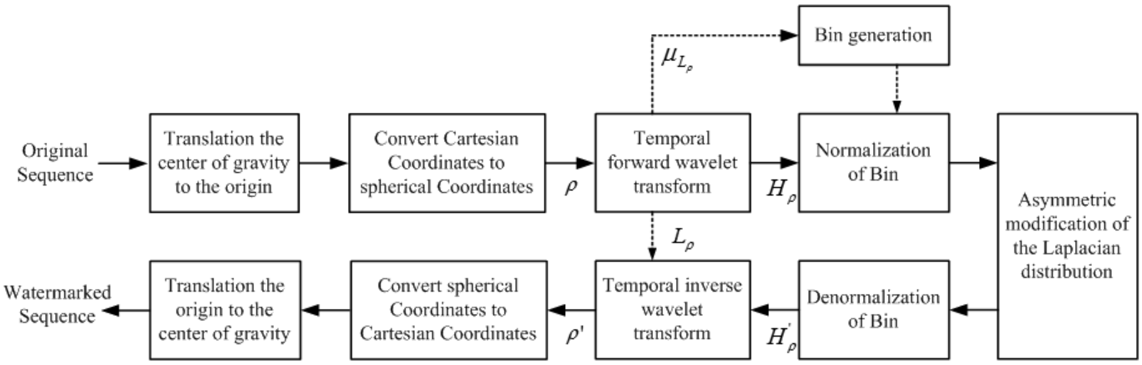

Bin generation
μ_{L_ρ}
Original Sequence
Translation the center of gravity to the origin
Convert Cartesian Coordinates to spherical Coordinates
ρ
Temporal forward wavelet transform
H_ρ
Normalization of Bin
Asymmetric modification of the Laplacian distribution
L_ρ
Watermarked Sequence
Translation the origin to the center of gravity
Convert spherical Coordinates to Cartesian Coordinates
ρ'
Temporal inverse wavelet transform
H'_ρ
Denormalization of Bin

Figure 6.6: Block diagram of the watermark embedding procedures of the second proposal

Before ganerating the bins, the average of low frequency coefficients $\mu_{s_\rho}(i)$ is calculated by

$$\mu_{l_\rho}(i) = \frac{1}{|S|/2} \sum_{t=0}^{|S|/2-1} l_t(i) \tag{6.11}$$

The average frame, $\mu_{l_\rho}(i)$ $(0 \leq i < |v|)$ is divided into N distinct bins with equal range, according to their magnitude. The vertex index, i in n-th bin of $\mu_{l_\rho}(i)$, will be kept to modify their corresponding high frequency coefficients, $h_{t,n}(i)$. High frequency coefficients belonging to n-th bin, $h_{t,n}(i)$, are mapped into normalized range of $[-1, 1]$ by Eq. (6.3).

Now, the distribution of the normalized high frequency coefficients $\widetilde{h}_{t,n}(i)$ has Laplacian distribution with symmetric property in the range of $[-1, 1]$. High frequency coefficients are divided into two groups, positive and negative side, by the sign of $\widetilde{h}_{t,n}(i)$. To embed n-th watermark bit $\omega_n \in \pm 1$, the variance of $\widetilde{h}_{t,n}(i)$ is modified by Eq. (6.7).

To modify the variance to the desired level, high frequency coefficients, $\tilde{s}_{2t+1,n}(i)$, are transformed iteratively by a histogram mapping function given by Eq. (6.8).

All transformed high frequency coefficients in each bin are mapped onto the original range. Then, temporal inverse wavelet transform is performed and the all sequences are translated the origin to the original center of gravity (μ_{x_t}, μ_{y_t}, μ_{z_t})to get the watermarked 3-D dynamic mesh sequence.

$$\begin{aligned} x_t^{'}(i) &= \rho_t(i)\text{cos}\theta_t(i)\text{sin}\phi_t(i) + \mu_{x_t} \\ y_t^{'}(i) &= \rho_t(i)\sin\theta_t(i)\cos\phi_t(i) + \mu_{y_t} \\ z_t^{'}(i) &= \rho_t(i)\cos\phi_t(i) + \mu_{z_t} \end{aligned} \tag{6.12}$$

The extraction procedure is shown in Fig. 6.7. Similar to the watermark embedding process, the center of gravity of the watermarked 3-D mesh sequence S'' is translated to the origin. Cartesian coordinates of the vertices of $s_t^{''}$ is converted to spherical coordinates. Temporal wavelet transform is applied to the vertex norms of each frames. The vertex norms in the average of low frequency coefficients $l_t^{''}$ are divided into N distinct bins with

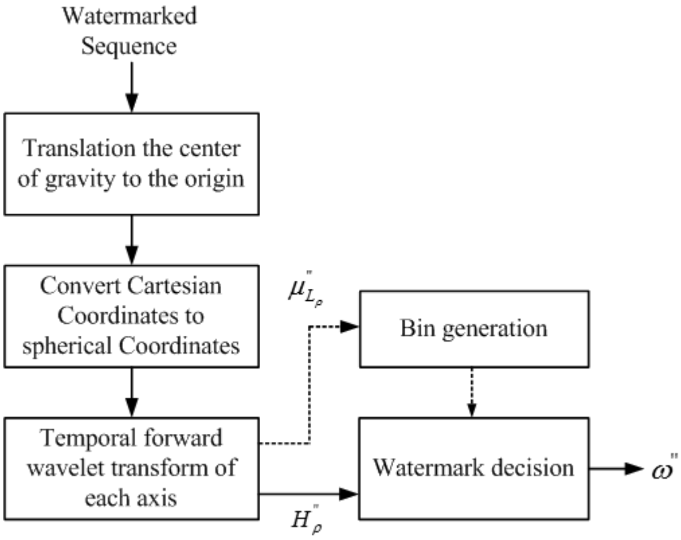

Watermarked Sequence
Translation the center of gravity to the origin
Convert Cartesian Coordinates to spherical Coordinates
Temporal forward wavelet transform of each axis
μ''_{L_ρ}
Bin generation
Watermark decision
H''_ρ
ω''

Figure 6.7: Block diagram of the watermark extraction procedures of the second proposal

equal range, according to their magnitude. The high frequency coefficients belonging to n-th bin, $h''_{t,n}(i)$, are mapped into normalized range of $[-1, 1]$. The normalized high frequency coefficients $\widetilde{h}''_{t,n}(i)$ in each bin are grouped into positive and negative side according to the sign of $\widetilde{h}''_{t,n}(i)$. The variance of positive and negative side, $E[Y^{2''}_{n,+}]$ and $E[Y^{2''}_{n,-}]$, of n-th bin, is calculated and compared each other to detect the watermark ω_n by Eq. (6.9). Note that the watermark extraction process does not require the original mesh sequence.

6.3.2 Modification of Both Low and High Frequency Coefficients of Vertex Norms

In Section 6.2 and 6.3.1, we have proposed two blind watermarking methods for 3-D mesh sequences. The methods used temporal wavelet transform and the variance high frequency coefficients is asymmetrically modified. The first method (Section 6.2) used three axes of x, y and z. The advantage of the first approach is that the high capacity, however, the robustness against rotation attack and the invisibility of the watermark could negatively effected. The second approach (Section 6.3.1) is able to overcome the problem by using only one axis of vertex norm ρ for the watermark embedding. Although the capacity and the robustness of the watermark could be controlled by the watermark strength factor, the capacity is limited by the fact that the bin generation needs to be done by the average of the low frequency coefficients. As it has been investigated in Section 3.1, both low and high frequency coefficients are able to be the watermark embedding primitives from the property of the wavelet decomposition. In this subsection, we propose the watermarking method by modifying the distribution of both low and high frequency coefficients. The method utilized our previous methods introduced in [CHO07] and in Section 6.3.1. Note that the extraction

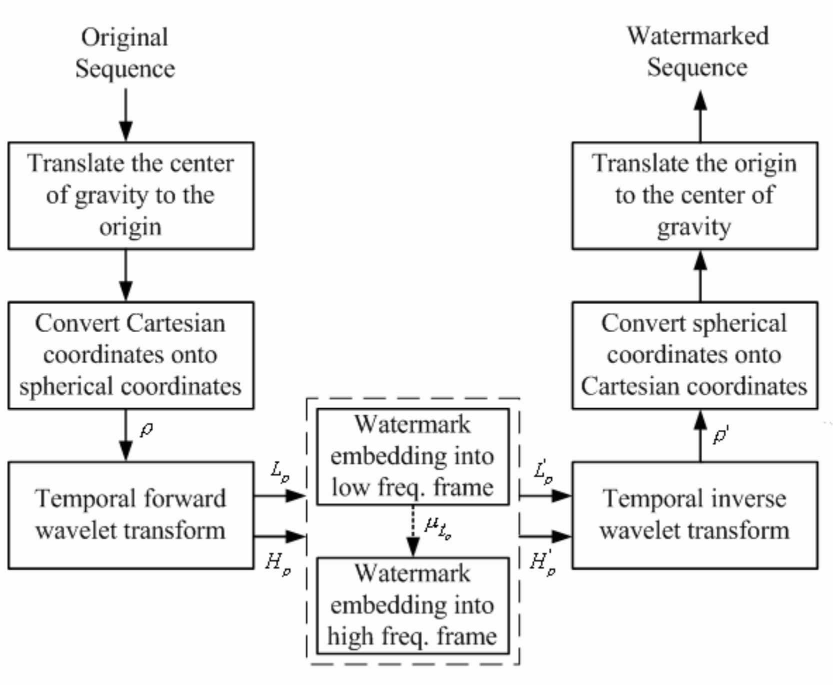

Original Sequence
Translate the center of gravity to the origin
Convert Cartesian coordinates onto spherical coordinates
ρ
Temporal forward wavelet transform
L_ρ
H_ρ
Watermark embedding into low freq. frame
μ_{L_ρ}
Watermark embedding into high freq. frame
L'_ρ
H'_ρ
Temporal inverse wavelet transform
ρ'
Convert spherical coordinates onto Cartesian coordinates
Translate the origin to the center of gravity
Watermarked Sequence

Figure 6.8: Watermark Embedding

of watermark of each low frequency frame can be done by directly using the variance of vertex norms distribution of the single frame without temporal wavelet transform.

Fig. 6.8 shows the watermark embedding procedures. From the original sequence S, the vertices of each frame s_t in Cartesian coordinates are converted onto spherical coordinates followed by the center of gravity of each frame is translated to the origin using Eq. (6.11).

The vertex norm $\rho_t(i)$ is only decomposed into low frequency coefficients $l_t(i)$ and high frequency coefficients $h_t(i)$ by temporal wavelet transform. The watermark is embedded into the low frequency frames $l_t(i)$ as well as into the high frequency coefficients $h_t(i)$.

The variance shifting method introduced in Section 5.2.2 is applied for watermark embedding at each low frequency frames $l_t(i)$, where $0 \leq i < |V_t|$ and $|V_t|(= |V|)$ is the number of vertices. For the simplicity, let ρ_i is the i-th vertex norm of the each low frequency frame. The histogram of ρ_i is divided into N distinct bins with equal range according to their magnitude. The r-th vertex norms in n-th bin, $\rho_{n,r}$, are all mapped into the range of $[-1, 1]$. We assume that each bin has a uniform distribution over the interval $[-1, 1]$.

In the variance modification step, the variance of n-th bin, σ_n^2, is modified by Eq. (5.10). To modify the variance to the desired level, the normalized vertex norms $\tilde{\rho}_{n,r}$, are transformed iteratively by a histogram mapping function as given by Eq. (5.11). The watermark embedding at each low frequency frame is finished by mapping the transforming vertex norms $\tilde{\rho}'_{n,r}$ of each bin onto the original range.

Since we only use the vertex norms for the temporal wavelet transform, the vector norm modification method in Section 6.3 is utilized for the watermark embedding at high frequency coefficients. Note that the average frame is calculated by the watermarked low frequency coefficients $s_{2t}^{'}(i)$ by

$$\mu_{s_{\rho}^{'}}(i) = \frac{1}{|S|/2} \sum_{t=0}^{|S|/2-1} s_{2t}^{'}(i) \tag{6.13}$$

The average frame, $\mu_{s_{\rho}^{'}}(i)$ is divided into N distinct bins with equal range according to their magnitude. The vertex index, i in n-th bin of $\mu_{s_{\rho}}(i)$, will be kept to modify their corresponding high frequency coefficients, $s_{2t+1,n}(i)$. High frequency coefficients belonging to n-th bin, $\tilde{s}_{2t+1,n}(i)$, are mapped into normalized range of $[-1, 1]$ by Eq. (6.3).

Now, the distribution of the normalized high frequency coefficients $\tilde{s}_{2t+1,n}(i)$ has Laplacian distribution with symmetric property in the range of $[-1, 1]$. High frequency coefficients are divided into two groups, positive and negative side, by the sign of $\tilde{s}_{2t+1,n}(i)$. To embed n-th watermark bit $\omega_n \in \pm 1$, the variance of $\tilde{s}_{2t+1,n}(i)$ is modified by Eq. (6.7).

To modify the variance to the desired level, high frequency coefficients, $\tilde{s}_{2t+1,n}(i)$, are transformed iteratively by a histogram mapping function given by Eq. (6.8).

All transformed high frequency coefficients in each bin are mapped onto the original range. Then, temporal inverse wavelet transform is applied to the watermarked low frequency coefficients $s_{2t}^{'}(i)$ and the high frequency coefficients $s_{2t+1}^{'}(i)$. Each watermarked frame $s_t^{'}$are translated the origin to the original center of gravity (μ_{x_t}, μ_{y_t}, μ_{z_t})to get the watermarked 3-D dynamic mesh sequence.

Fig. 6.9 shows the watermark extraction procedures of the proposed method.

From the watermarked 3-D mesh sequence S'', The center of gravity of each frame is translated to the origin. Cartesian coordinates of the vertices of $s_t^{''}$ is converted to spherical coordinates. Temporal wavelet transform is applied to the vertex norms of each frames.

Following Section 5.2.2, the variance of each bin, $\sigma_n^{2''}$, is calculated and compared with the threshold value, $\frac{1}{3}$. The watermark hidden in the n-th bin, ω_n'', is extracted by Eq. (5.13). Note that the extraction of watermark of each low frequency frame can be done by directly using the variance of vertex norms distribution of the single frame without temporal wavelet transform.

To extract the watermark which is embedded in the high frequency coefficients, the vertex norms in the average frame $\mu_{s_{\rho}^{''}}(i)$ are divided into N distinct bins with equal range according to their magnitude. The high frequency coefficients belonging to n-th bin, $h_{t,n}^{''}(i)$, are mapped into normalized range of $[-1, 1]$. The normalized high frequency coefficients $\widetilde{h}_{t,n}^{''}(i)$ in each bin are grouped into positive and negative side according to the sign of $\widetilde{h}_{t,n}^{''}(i)$. The variance of positive and negative side, $E[Y_{n,+}^{2''}]$ and $E[Y_{n,-}^{2''}]$, of n-th bin, is calculated and compared each other to detect the watermark ω_n by Eq. (6.9). Note that the watermark extraction process does not require the original mesh sequence.

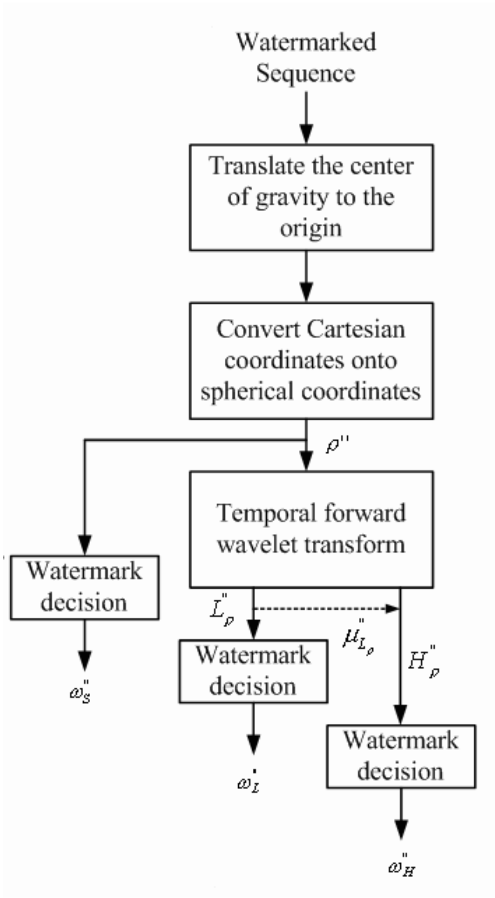
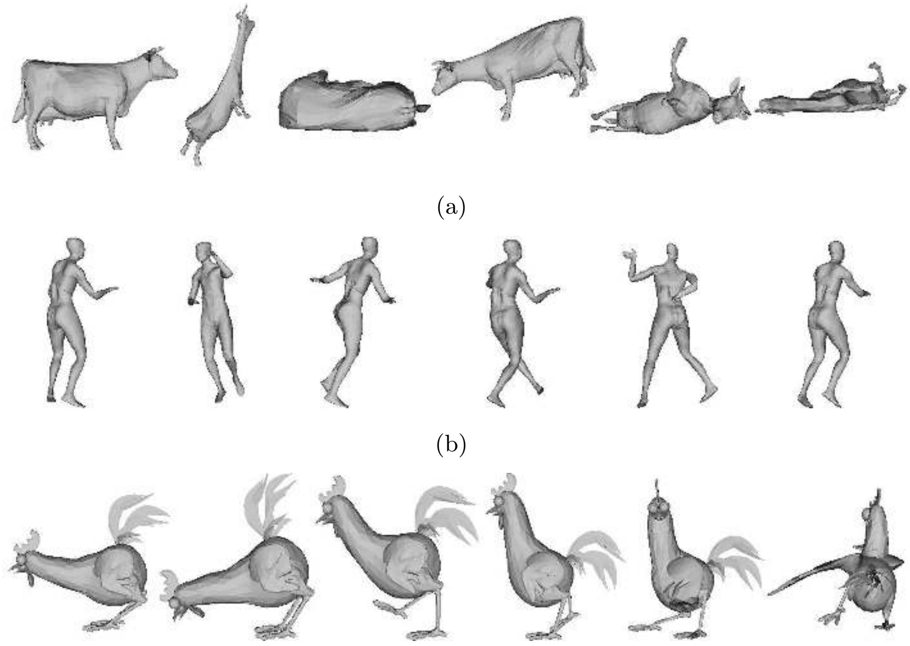


Figure 6.9: Watermark Extraction

6.4 Experimental Results

In this section, we show the experimental results of the proposed methods. The experimental results are given in two subsections. In Section 6.4.1, we evaluate the the invisibility and the robustness of watermarking methods which modify only high frequency coefficients. Note that the first method (Section 6.2) embeds the watermark into three axes of x, y and z and the second method (Section 6.3.1) embeds the watermark into one axis of vertex norm ρ. For the notational simplicity, we denoted the first method as 'Method I' and the second as 'Method II'. We evaluate the performance of the combined method ('Method III') in Section 6.3.2. We conducted experiments on three sequences shown in Fig. 6.10: Cow, Dance, and Chicken. The characteristics of the test sequences are listed in Table 6.1. Dyadic 5/3-tap bi-orthogonal perfect reconstruction filter bank is uses for temporal wavelet decomposition. The quality is measured by the maximum root mean square distance E_{mrms} of each static mesh frames. We calculated the average of E_{mrms} of all static mesh frames, and denoted as $E_{avg}(V, V')$. In addition, the robustness of the watermark is evaluated by correlation coefficient, $corr$, between the designed and extracted watermark by Eq. (4.5).



(a)

(b)

(c)

Figure 6.10: Test sequences (a) Cow, (b) Dance, and (c) Chicken

Table 6.1: The characteristics of test mesh sequences

	Cowheavy	Dance	Chicken
Number of frames	204	200	396
Number of vertices	2,904	7,061	2,916
Number of faces	5,804	14,118	5,454
Number of edges	8,706	21,177	8,326
Number of internal edges	8,706	21,177	8,036
Number of boundary edges	0	0	290
Number of holes (boundaries)	0	0	30
Number of connected components	1	1	37
Manifold	No	Yes	No
Oriented	No	Yes	No
Mesh Type	Irregular	Irregular	Irregular

6.4.1 Modification of High Frequency Coefficients

In this subsection, we show the performance of Method I and II. In the experiment, the wavelet decomposition levels is 2 and the watermark is embedded into high frequency coefficients and into middle frequency coefficients, respectively, as an example.

Table 6.2: Evaluation of the invisibility of Method I and II with the different strength factors of α when no attack.

Embedding Level	Model	α	Method I	Method II
			$E(V,V')$	$E(V,V')$
High Frequency	Cow	0.04	0.45×10^{-4}	0.22×10^{-4}
		0.08	0.58×10^{-4}	0.29×10^{-4}
		0.10	0.66×10^{-4}	0.33×10^{-4}
	Dance	0.04	0.26×10^{-4}	0.05×10^{-4}
		0.08	0.32×10^{-4}	0.06×10^{-4}
		0.10	0.36×10^{-4}	0.07×10^{-4}
	Chicken	0.04	6.58×10^{-4}	2.42×10^{-4}
		0.08	8.14×10^{-4}	2.99×10^{-4}
		0.10	8.96×10^{-4}	3.29×10^{-4}
Middle Frequency	Cow	0.04	0.87×10^{-4}	0.56×10^{-4}
		0.08	1.20×10^{-4}	0.75×10^{-4}
		0.10	1.37×10^{-4}	0.85×10^{-4}
	Dance	0.04	0.60×10^{-4}	0.12×10^{-4}
		0.08	0.75×10^{-4}	0.15×10^{-4}
		0.10	0.83×10^{-4}	0.17×10^{-4}
	Chicken	0.04	10.60×10^{-4}	3.88×10^{-4}
		0.08	13.22×10^{-4}	4.89×10^{-4}
		0.10	14.62×10^{-4}	5.43×10^{-4}

To evaluate the invisibility of the proposed methods, we embedded the watermark into high (or middle) frequency coefficients with different strength factors *alpha*. The watermark capacity is 64 bits per an axis. Note that total of 192 bits are embedded by Method I. Table 6.2 lists $E_{avg}(V, V')$, the average of E_{mrms} of each frame. The visual distortion increases in proportion to the strength factor *alpha*. The invisibility of the watermark in high frequency domain is better than that in middle frequency domain. When the invisibility of two methods looked at each other, Method II shows better results than Method I. The main reason can be explained by the number of the embedded watermark bits. Note that Method I embeds three times more watermark bits than Method II.

In Fig. 6.11, the invisibility of the watermark of each proposed method is compared by 'Chicken' sequence. The figure shows the 262-th frame which has the biggest motion in the sequence. The strength factor of $\alpha = 0.08$ is applied for both methods. When the original Chicken (Fig. 6.11 (a)) are compared with the watermarked at high frequency domain (Fig. 6.11 (b) and Fig. 6.11 (c)), both of two methods are almost invisible to human eyes. However, Method I (Fig. 6.11 (d)) shows visible distortions in the case of the embedding into middle frequency coefficients while the watermark embedded by Method II (Fig. 6.11 (e)) is still transparent.

We also evaluated the robustness of the watermark of the proposed methods in the case of no attack and listed in Table 6.3. The numbers of the watermark are the same as Table

Table 6.3: Evaluation of the robustness of the proposed methods in terms of the different watermark strength factors of α when no attck

Embedding Level	Model	α	Method I				Method II
			$corr_x$	$corr_y$	$corr_z$	$corr_{avg}$	$corr$
High frequency	Cow	0.04	1.00	0.83	1.00	0.94	1.00
		0.08	1.00	1.00	1.00	1.00	1.00
		0.10	1.00	1.00	1.00	1.00	1.00
	Dance	0.04	1.00	0.97	1.00	0.99	1.00
		0.08	1.00	1.00	1.00	1.00	1.00
		0.10	1.00	1.00	1.00	1.00	1.00
	Chicken	0.04	1.00	1.00	0.97	0.99	1.00
		0.08	1.00	1.00	1.00	1.00	1.00
		0.10	1.00	1.00	1.00	1.00	1.00
Middle frequency	Cow	0.04	1.00	0.83	1.00	0.94	1.00
		0.08	1.00	1.00	1.00	1.00	1.00
		0.10	1.00	1.00	1.00	1.00	1.00
	Dance	0.04	1.00	1.00	1.00	1.00	1.00
		0.08	1.00	1.00	1.00	1.00	1.00
		0.10	1.00	1.00	1.00	1.00	1.00
	Chicken	0.04	1.00	1.00	1.00	1.00	0.85
		0.08	1.00	1.00	1.00	1.00	0.97
		0.10	1.00	1.00	1.00	1.00	0.97

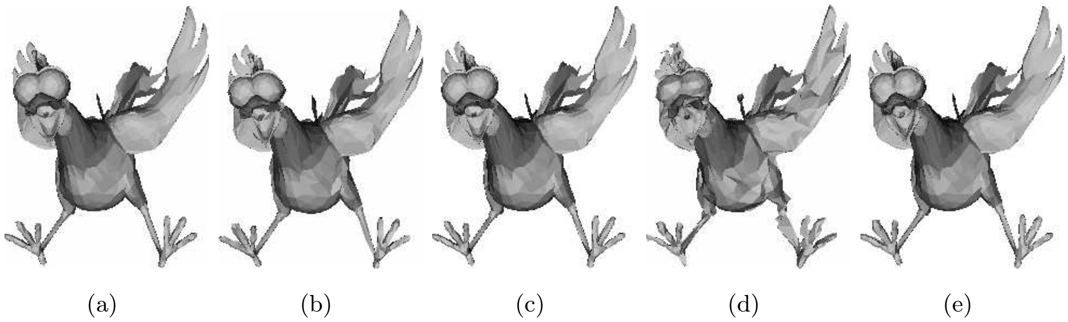
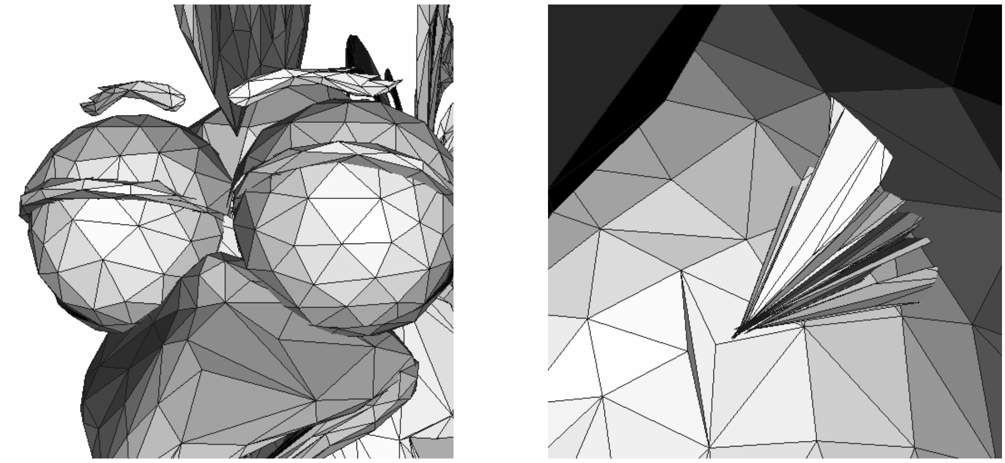


Figure 6.11: 262-th frame of Chicken sequences watermarked with the strength factor of $\alpha = 0.08$: (a) The original, (b) high frequency domain, Method I, (c) high frequency domain, Method II, (d) middle frequency domain, Method I, and (e) middle frequency domain, Method II

6.2. Method I showed error-free extraction when the strength factor α is more than 0.08 in both high and middle frequency domain.

In Method II, Cow and Dance showed error-free extraction, but Chicken shows relatively smaller in the case of the watermark in the middle frequency domain. This is caused by both the number of connected components in the frame and the structure of edges of the meshes.

Each frame of Chicken has 37 connected components while Cow and Dance have a single number of connected components (see Table 6.1). A mesh model like Chicken makes it difficult to find the exact center of gravity in the watermark extraction procedures. Non-manifold mesh structure of Chicken is also could be a problem to find the center of gravity. Fig. 6.12 shows non-manifold meshes of Chicken, as example. As the bin is generated by low frequency coefficient norm μ_{l_ρ}, the exact center of gravity as the original meshes is required for both methods to extract the error-free watermark in the case of a meshes like 'Chicken'.



(a) (b)

Figure 6.12: Non-manifold meshes (a) eye-brows of Chicken (b) inner side of Chicken

Table 6.4: Evaluation of the capacity and invisibility of the Method II with fixed strength factor $\alpha = 0.08$ in the case of no attack where embedding at high frequency domain

Model	N	$corr$	$E_{avg}(V, V')$
Cow	32	1.00	0.27×10^{-4}
	64	1.00	0.29×10^{-4}
	128	1.00	0.29×10^{-4}
	256	0.96	0.29×10^{-4}
Dance	32	1.00	0.07×10^{-4}
	64	1.00	0.06×10^{-4}
	128	1.00	0.06×10^{-4}
	256	0.99	0.06×10^{-4}
Chicken	32	1.00	2.81×10^{-4}
	64	1.00	2.99×10^{-4}
	128	0.88	3.83×10^{-4}
	256	0.85	3.21×10^{-4}

Method II uses only one axis of vertex norms as watermark carrier. Clearly, the capacity of Method II is less than Method I. To evaluate the performance in terms of the watermark capacity, different numbers of the watermark are embedded with the fixed strength factor

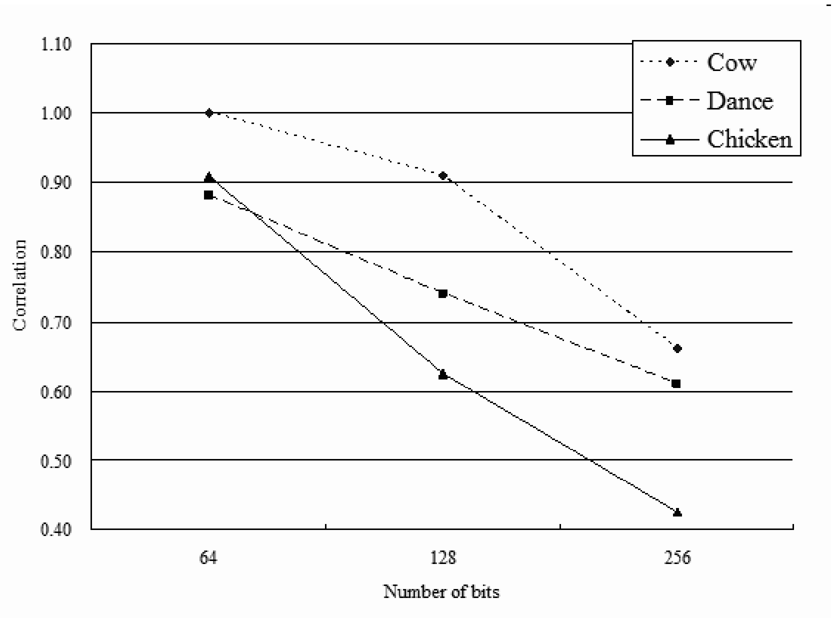


Figure 6.13: Evaluation of the capacity and robustness of Method II with fixed strength factor ($\alpha = 0.08$) in the case of uniform coordinate quantization (7 bits) attack

$alpha = 0.08$. Table 6.4 and Fig. 6.13 list the the invisibility and the robustness against uniform quantization attack of 7 bits, respectively. When the watermark capacity is increased, $E_{avg}(V, V')$ also increases but the rate is so small as to be ignored in the case of 'Cow and Dance' comparing with the artifact by the strength factor in Table 6.2. Fig. 6.4, the robustness in terms of the capacity is evaluated by uniform coordinates quantization of 7 bits. As increasing the capacity, the robustness of the watermark decreases. The figure shows that there is trade off between the robustness and the capacity of Method II.

We evaluate the performances of the robustness against several attacks with the strength factor $\alpha = 0.08$ as example. Additive noise, Laplacian smoothing, uniform quantization, frame averaging and rotation attack are applied, respectively.

Table 6.5 shows the robustness against additive noise attacks. For the noise attacks, binary random noise was added to each vertex coordinate in each frame with three different error rates: 0.1%, 0.3%, and 0.5%. Fairly good robustness can be expected for the error rate less than the error rate of 0.3% for the watermark into high frequency coefficients and the error rate 0.5% for the watermark at middle frequency domain. The robustness of the watermark embedded in Chicken is not so good when embedding at middle frequency coefficients caused by the center of the gravity.

Table 6.6 shows the robustness against smoothing attacks. The smoothing attacks are carried out by using Laplacian smoothing [FIEL88]. We applied Laplacian smoothing with the relaxation factor of 0.03 and the iteration 10, 30 and 50 times each. The robustness against smoothing attack is related to the curvatures of the mesh model. For example, less

Table 6.5: Evaluation of the robustness against additive noise attacks of the proposed methods where $\alpha = 0.08$

Embedding level	Model	Error rate	Method I				Method II
			$corr_x$	$corr_y$	$corr_z$	$corr_{avg}$	$corr$
High frequency	Cow	0.1%	1.00	1.00	1.00	1.00	1.00
		0.3%	1.00	0.85	1.00	0.95	1.00
		0.5%	1.00	0.91	0.94	0.95	1.00
	Dance	0.1%	1.00	0.91	1.00	0.97	1.00
		0.3%	1.00	0.94	1.00	0.98	0.97
		0.5%	1.00	0.88	1.00	0.96	0.88
	Chicken	0.1%	1.00	0.94	1.00	0.98	1.00
		0.3%	1.00	0.94	1.00	0.98	1.00
		0.5%	1.00	0.91	1.00	0.97	0.97
Middle frequency	Cow	0.1%	1.00	1.00	1.00	1.00	1.00
		0.3%	1.00	0.97	1.00	0.99	1.00
		0.5%	1.00	0.97	1.00	0.99	0.97
	Dance	0.1%	1.00	1.00	1.00	1.00	1.00
		0.3%	1.00	0.91	1.00	0.97	1.00
		0.5%	1.00	0.83	1.00	0.94	1.00
	Chicken	0.1%	0.66	0.94	0.63	0.74	0.70
		0.3%	0.69	0.91	0.66	0.75	0.83
		0.5%	0.85	0.91	0.59	0.78	0.85

Table 6.6: Evaluation of the robustness against Laplacian smoothing attacks of the proposed methods where $\alpha = 0.08$

Embedding level	Model	Iteration	Method I				Method II
			$corr_x$	$corr_y$	$corr_z$	$corr_{avg}$	$corr$
High frequency	Cow	10	1.00	0.77	0.97	0.91	1.00
		30	0.78	0.55	0.77	0.70	0.44
		50	0.60	0.53	0.55	0.56	0.41
	Dance	10	1.00	0.94	1.00	0.98	1.00
		30	0.97	0.80	0.84	0.87	0.94
		50	0.75	0.75	0.66	0.72	0.72
	Chicken	10	0.69	0.72	0.11	0.51	0.44
		30	0.38	0.54	0.00	0.31	0.16
		50	0.19	0.16	-0.07	0.09	0.19
Middle frequency	Cow	10	1.00	0.75	0.94	0.90	1.00
		30	0.75	0.55	0.67	0.66	0.60
		50	0.60	0.33	0.43	0.46	0.44
	Dance	10	1.00	1.00	1.00	1.00	1.00
		30	0.81	0.74	0.88	0.81	0.91
		50	0.69	0.71	0.66	0.69	0.73
	Chicken	10	0.50	0.79	0.57	0.62	0.49
		30	0.19	0.35	0.16	0.23	0.29
		50	0.06	0.25	0.06	0.13	0.18

Table 6.7: Evaluation of the robustness against uniform quantization attacks of the proposed methods where $\alpha = 0.08$

Embedding level	Model	Bits	Method I				Method II
			$corr_x$	$corr_y$	$corr_z$	$corr_{avg}$	$corr$
High frequency	Cow	9	1.00	1.00	1.00	1.00	1.00
		8	1.00	0.83	0.97	0.93	1.00
		7	1.00	0.80	0.97	0.92	1.00
	Dance	9	1.00	0.91	1.00	0.97	1.00
		8	1.00	0.88	1.00	0.96	1.00
		7	1.00	0.63	1.00	0.88	0.88
	Chicken	9	0.79	0.88	0.75	0.81	0.94
		8	0.76	0.94	0.80	0.83	1.00
		7	0.82	0.91	0.82	0.85	0.91
Middle frequency	Cow	9	1.00	0.97	1.00	0.99	1.00
		8	1.00	0.97	1.00	0.99	1.00
		7	1.00	0.91	1.00	0.97	1.00
	Dance	9	1.00	1.00	1.00	1.00	1.00
		8	1.00	0.94	1.00	0.98	1.00
		7	1.00	0.83	1.00	0.94	1.00
	Chicken	9	0.56	0.94	0.47	0.66	0.74
		8	0.66	0.97	0.47	0.70	0.72
		7	0.72	0.97	0.65	0.78	0.65

curvatures of Dance sequence make the watermark more robust than the other sequences in both of ours methods in high and middle frequency domain. When the iteration is more than 30 times, Chicken shows relatively lower robustness against Laplacian smoothing. The neighboring vertices are more centralized to pivot vertex by a larger number of iterations. It means that some separated parts from the body of Chicken could be shrunk by Laplacian smoothing. This artifact comes to the center of gravity problem.

In Table 6.7, each vertex coordinate is quantized to 9, 8 and 7 bits for uniform quantization attacks. The watermark in the middle frequency domain is robust against uniform quantization attacks in the case of Cow and Dance sequences. Chicken shows better robustness only in high frequency domain caused by the inaccurate center of gravity.

To evaluate the robustness against temporal de-synchronization, we applied frame-averaging attacks to the watermarked sequences. We dropped one frame for each 24, 12 and 6 frames from the watermarked sequence and the missing frame was replaced with the interpolation using two adjacent frames. As we have been expected, Table 6.8 shows the fairly good robustness against frame averaging attacks since our methods uses temporal wavelet transform. The watermark in middle frequency domain showed better *corr* than in high frequency domain.

For rotation attacks, each frame of the watermarked sequences are rotated by two different angles, $(1°, 2°, 3°)$ and $(15°, 20°, 30°)$, respectively. Because each axis of the wavelet

Table 6.8: Evaluation of the robustness against frame averaging attacks of the proposed methods where $\alpha = 0.08$

Embedding level	Model	$\frac{1}{frames}$	Method I				Method II
			$corr_x$	$corr_y$	$corr_z$	$corr_{avg}$	$corr$
High frequency	Cow	24	1.00	1.00	1.00	1.00	0.91
		12	0.68	0.97	0.97	0.87	0.84
		6	0.52	0.82	0.78	0.71	0.85
	Dance	24	1.00	1.00	1.00	1.00	0.88
		12	1.00	1.00	1.00	1.00	0.85
		6	0.80	0.85	0.91	0.86	0.13
	Chicken	24	0.88	0.97	1.00	0.95	0.70
		12	0.88	0.97	1.00	0.95	0.59
		6	0.75	0.71	0.41	0.62	0.45
Middle frequency	Cow	24	1.00	1.00	1.00	1.00	1.00
		12	1.00	1.00	1.00	1.00	1.00
		6	1.00	0.97	1.00	0.99	1.00
	Dance	24	1.00	1.00	1.00	1.00	1.00
		12	1.00	1.00	1.00	1.00	1.00
		6	1.00	1.00	1.00	1.00	1.00
	Chicken	24	0.81	0.97	0.80	0.86	0.94
		12	0.81	0.97	0.77	0.85	0.94
		6	0.81	0.94	0.70	0.82	0.88

Table 6.9: Evaluation of the robustness against rotation attacks of the proposed methods where $\alpha = 0.08$

Embedding level	Model	$(x^\circ, y^\circ, z^\circ)$	Method I				Method II
			$corr_x$	$corr_y$	$corr_z$	$corr_{192}$	$corr_{64}$
High frequency	Cow	$(1, 2, 3)$	1.00	1.00	1.00	1.00	1.00
		$(15, 20, 13)$	1.00	0.65	1.00	0.88	1.00
	Dance	$(1, 2, 3)$	1.00	1.00	1.00	1.00	1.00
		$(15, 20, 13)$	1.00	-0.04	1.00	0.65	1.00
	Chicken	$(1, 2, 3)$	0.97	0.97	0.35	0.76	1.00
		$(15, 20, 13)$	0.59	0.78	0.16	0.51	1.00
Middle frequency	Cow	$(1, 2, 3)$	1.00	0.97	1.00	0.99	1.00
		$(15, 20, 13)$	0.97	0.77	0.74	0.83	1.00
	Dance	$(1, 2, 3)$	1.00	1.00	1.00	1.00	1.00
		$(15, 20, 13)$	0.94	0.16	0.91	0.67	1.00
	Chicken	$(1, 2, 3)$	0.78	0.94	0.70	0.81	0.97
		$(15, 20, 13)$	0.28	0.25	0.40	0.31	0.97

coefficients is modified independently, Method I is not robust against heavy rotation. As the vector norm which is invariant to rotation, Method II can perfectly extract the hidden watermark in the case of Cow and Dance. Chicken watermarked by Method II has the same correlation coefficients as is in the case of no attack.

6.4.2 Modification of Both Low and High Frequency Coefficients

To evaluate the performance of the Method III, the different watermark bits are embedded into each low frequency frame and into high frequency coefficients after 1 level wavelet decomposition, as example. N^L bits of the watermark with α^L are embedded into each low frequency frames, independently, and N^H bits with α^H are embedded into high frequency band.

The robustness of watermark is measured by correlation coefficients (Eq. (4.5)). Note that the watermark of low frequency frame N^L is extracted from each low frequency frames so that the average correlation coefficient is calculated and is denoted as $corr^L_{avg}$. $corr^H$ indicates the correlations coefficients from the watermark of high frequency band. In particular, the watermark in each low frequency frame is also able to extract in spatial domain and is indicated as $corr^S_{avg}$.

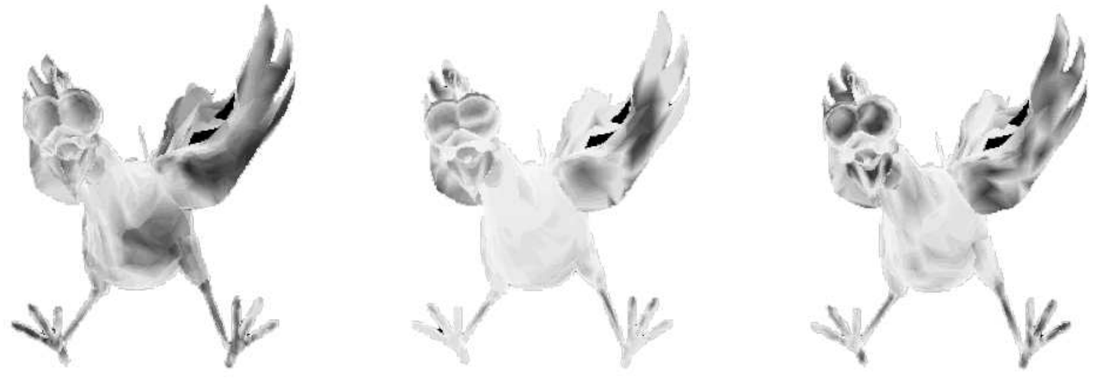
Table 6.10: Evaluation of the invisibility and the robustness of the watermark with different strength factors when no attack

	α	Δ	$corr^S_{avg}$	$corr^L_{avg}$	$corr^H$	$E_{avg}(V,V')$
Cow	0.04	0.04	0.41	1.00	1.00	0.44×10^{-4}
	0.04	0.08	0.41	1.00	1.00	0.49×10^{-4}
	0.08	0.08	0.50	1.00	1.00	0.71×10^{-4}
Dance	0.04	0.04	0.70	1.00	1.00	0.37×10^{-4}
	0.04	0.08	0.70	1.00	1.00	0.38×10^{-4}
	0.08	0.08	0.76	1.00	1.00	0.69×10^{-4}
Chicken	0.04	0.04	0.67	0.97	0.91	8.45×10^{-4}
	0.04	0.08	0.67	0.97	1.00	8.91×10^{-4}
	0.08	0.08	0.74	0.97	1.00	12.74×10^{-4}

We first evaluate the performance of the invisibility and the robustness when embedding the watermark with different strength factors. Here, 32 bits of the watermark are embedded into each low frequency frames and 64 bits are embedded into high frequency band as example. Table 6.10 lists the results when no attack. As expected, the robustness is increased as the strength factor is increased, however, the invisibility of the watermark is decreases according to the strength factors. In addition, the visual distortion $E_{avg}(V,V')$ of Method III is slightly increased when comparing with that of the Method I and II in Section 6.4.1.

Fig. 6.14 shows the 262-th frame of the watermarked 'Chicken' sequence which has the biggest motion in the sequence. This figure depicts that Method III (Fig. 6.14-(c))introduces more visual distortion than Method II (Fig. 6.14-(b)), however, that distortion is relatively small when comparing with Method I (Fig. 6.14-(a)).

As we mentioned in Section 6.3.2, the hidden watermark in each low frequency frames can be also extracted in any frame in the original sequence if we assume that the same



(a) Method I, $\alpha^H = 0.08$, $N^H = 192$, $E_{mrms} = 57.96 \times 10^{-4}$

(b) Method II, $\alpha^H = 0.08$, $M = 64$, $E_{mrms} = 21.87 \times 10^{-4}$

(c) Method III, $\alpha^L = 0.08$, $B = 32$, $\alpha^H = 0.08$, and $M = 64$, $E_{mrms} = 28.75 \times 10^{-4}$

Figure 6.14: 262-th frame of Chicken sequence watermarked by the proposed methods. Dark region indicates the distorted one caused by watermark embedding. Visual distortion E_{mrms} by Method III is less than that by Method I

watermark bits are embedded into each low frequency frames. In Table 6.10, $corr^S_{avg}$ indicates the average of correlation coefficients of each frames in spatial domain. The strength factor α^L increases the correlation coefficients $corr^S_{avg}$. Note that the visual distortion also increases in proportion to α^L. In addition, the distortion introduced by α^L is more than that introduced by α^H.

Comparing $corr^S_{avg}$ with $corr^L_{avg}$, $corr^S_{avg}$ is lower than $corr^S_{avg}$. This artifact could be explained by two reasons. The first reason is that, the high frequency coefficients is negatively effected like noises when the watermark extraction is performed at spatial domain. Another reason is the center of gravity. Our method needs to translate the center of gravity before watermark embedding and extraction. Then the probability distribution of the vector norm is divided to build the bins. Because the vector norm is very sensitive to the center of gravity, the center of gravity of the watermarked frame needs to be same with the original frame to build the exact bins. The modification of a low frequency frame l_n affects five frames $s_{n-2}, s_{n-1}, s_n, s_{n+1}, s_{n+1}$ in the case of 5/3-tap filter bank. Then our method is more suitable to the sequence that has relatively uniform movement through the whole frames. As results, when we compare $corr^S_{avg}$ of three test sequences, $corr^S_{avg}$ of Dance and Chicken shows better results than $corr^S_{avg}$ of Cow. Note that, Cow sequence has drastic movement between all adjacent frames.

Fig. 6.15 and Table 6.11 show the invisibility and the robustness of the proposed method evaluated with different number of embedded bits. Here, test sequence is Dance, strength factors of the watermarks are $\alpha^L = 0.08$, $\alpha^H = 0.08$. When the watermark capacity of low frequency frames N^L is increased, E_{mrms} decreases. The number of embedded bits N^H is

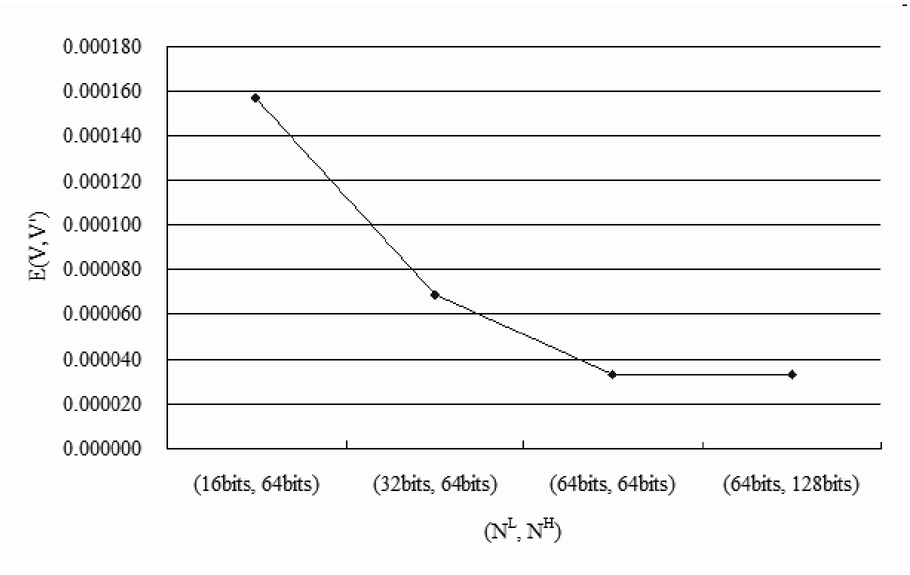


Figure 6.15: Evaluation of the invisibility of the proposed method with different number of embedded bits, at low frequency frames N^L, and at high frequency coefficients N^L, in the case of no attack. Here, test sequence is Dance, strength factors of the watermarks are $\alpha^L = 0.08$, $\alpha^H = 0.08$.

Table 6.11: Evaluation of the robustness of the proposed method with different number of embedded bits, at low frequency frames N^L, and at high frequency coefficients N^H, in the case of uniform coordinates quantization (7 bits). Here, test sequence is Dance, strength factors of the watermarks are $\alpha^L = 0.08$, $\alpha^H = 0.08$.

N^L	N^H	$corr^S_{avg}$	$corr^L_{avg}$	$corr^H$
16	64	0.93	1.00	0.91
32	64	0.66	0.95	0.84
64	64	0.19	0.51	0.90
64	128	0.18	0.34	0.88

relatively less affected the invisibility then N^L.

In Table 6.11, the robustness in terms of the capacity is evaluated by uniform coordinates quantization of 7 bits. The numbers of embedded bits of both low frequency frames N^L and high frequency coefficients N^H affect each correlation of extraction domain. There is a trade of between the capacity and the robustness. Note that, the less capacity of low frequency frame also decreases the invisibility of the watermark. When the range of the bin is widen, the more variation of the probability distribution could affect the more visual distortion.

To evaluate the robustness of the method, we applied several attacks to the watermarked sequence of $\alpha = 0.08$, $\alpha^H = 0.08$, $N^L = 32$ and $N^H = 64$. Additive noise, Laplacian smoothing, uniform coordinates quantization, frame averaging and rotation attack are applied, respectively. The robustness is also compared with Method I and Method II and

Table 6.12: Evaluation of the robustness against additive noise where the numbers of the embedded bits are $N^L = 32$ and $M = 64$ and the strength factors are $\alpha^L = 0.08$ and $\alpha^H = 0.08$.

Model	Error rate	Method III			Method I	Method II
		$corr^S_{avg}$	$corr^L_{avg}$	$corr^H$	$corr_{192}$	$corr_{64}$
Cow	0.3%	0.41	0.90	1.00	0.95	1.00
	0.5%	0.21	0.70	0.97	0.95	1.00
Dance	0.3%	0.75	0.97	0.87	0.98	0.97
	0.5%	0.65	0.93	0.91	0.96	0.88
Chicken	0.3%	0.63	0.85	0.97	0.98	1.00
	0.5%	0.48	0.72	0.90	0.98	1.00

Table 6.13: Evaluation of the robustness against Laplacian smoothing where the number of the embedded bits are $N^L = 32$ and $N^H = 64$, strength factors are $\alpha^L = 0.08$ and $\alpha^H = 0.08$.

Model	Iteration	Method III			Method I	Method II
		$corr^S_{avg}$	$corr^L_{avg}$	$corr^H$	$corr_{192}$	$corr_{64}$
Cow	30	0.05	0.14	0.78	0.70	0.44
	50	0.00	-0.01	0.61	0.56	0.41
Dance	30	0.51	0.71	1.00	0.87	0.94
	50	0.32	0.45	0.87	0.72	0.72
Chicken	30	-0.05	-0.05	0.29	0.31	0.16
	50	-0.05	-0.06	0.11	0.09	0.19

denoted as $corr_{192}$ and $corr_{62}$ where the subscript means the number of embedded bits. Note that, the capacity of the proposed method is $N^L \times \frac{1}{2}(thenumberofframes) + N^H$ bits.

Table 6.12 shows the robustness against additive noise attacks. For the noise attacks, binary random noise was added to each vertex coordinate in each frame with three different error rates: 0.3%, and 0.5%. Here, the error rate represents the amplitude of noise as a fraction of the maximum vertex norm of the object. Both $corr^L_{avg}$ and $corr^H_{avg}$ shows fairly good robustness against additive noise attack while embedding much more capacity than Method I and Method II.

Table 6.12 shows the robustness against smoothing attacks. The smoothing attacks are carried out by using Laplacian smoothing with the relaxation factor of 0.03 and the iteration 30 and 50 times each. Since the robustness against smoothing attack is related to the curvatures of the mesh model, Dance sequence is more robust than the other sequences. When $corr^H_{avg}$ is compared with Method I and Method II, the proposed method showed better robustness against Laplacian smoothing attack since the proposed method generate the bins from the average frame of the watermarked low frequency frames. In addition, the watermark in the high frequency coefficients is more robust against than the watermark in low frequency frame in the case of Cow and Dance sequence. However, the center of gravity

Table 6.14: Evaluation of the robustness against uniform coordinates quantization where the number of the embedded bits are $N^L = 32$ and $N^H = 64$, strength factors are $\alpha = 0.08$ and $\Delta = 0.08$.

Model	Bits	Method III			Method I	Method II
		$corr^S_{avg}$	$corr^L_{avg}$	$corr^H$	$corr_{192}$	$corr_{64}$
Cow	8 bits	0.45	0.94	1.00	0.93	1.00
	7 bits	0.34	0.80	1.00	0.92	1.00
Dance	8 bits	0.75	1.00	1.00	0.96	1.00
	7 bits	0.65	0.95	0.84	0.88	0.88
Chicken	8 bits	0.66	0.89	0.91	0.83	1.00
	7 bits	0.52	0.73	0.81	0.85	0.91

Table 6.15: Evaluation of the robustness against frame averaging quantization where the number of the embedded bits are $N^L = 32$ and $N^H = 64$, strength factors are $\alpha = 0.08$ and $\Delta = 0.08$.

Model	1/frames	Method III			Method I	Method II
		$corr^S_{avg}$	$corr^L_{avg}$	$corr^H$	$corr_{192}$	$corr_{64}$
Cow	12	0.50	0.97	0.87	0.87	0.84
	6	0.49	0.96	0.88	0.71	0.85
Dance	12	0.76	1.00	0.97	1.00	0.85
	6	0.76	1.00	0.32	0.86	0.13
Chicken	12	0.74	0.95	0.52	0.95	0.59
	6	0.74	0.93	0.24	0.62	0.45

problem still exists in the case of Chicken sequence after smoothing of 50 iterations.

In Table 6.14, each vertex coordinate is quantized to 8 and 7 bits for uniform quantization attacks. The proposed method has fairly good robustness against uniform quantization attack as shown in $corr^L$ and $corr^H$.

To evaluate the robustness against temporal de-synchronization, we applied frame-averaging attacks to the watermarked sequences. We dropped one frame for each 12 and 6 frames from the watermarked sequence and the missing frame was replaced with the interpolation using two adjacent frames. Table 6.15 shows the good robustness against frame averaging attacks both $corr^L_{avg}$ and $corr^H$. Note that, $corr^L_{avg}$ is better performance than $corr_{192}$ or $corr_{64}$.

Table 6.16: Evaluation of the robustness against rotation attack where the number of the embedded bits are $N^L = 32$ and $N^H = 64$, strength factors are $\alpha^L = 0.08$ and $\alpha^H = 0.08$.

Model	Degree	Method III			Method I	Method II
		$corr^S_{avg}$	$corr^L_{avg}$	$corr^H$	$corr_{192}$	$corr_{64}$
Cow	15°,20°,13°	0.50	1.00	1.00	0.88	1.00
Dance	15°,20°,13°	0.76	1.00	1.00	0.65	1.00
Chicken	15°,20°,13°	0.74	0.97	1.00	0.51	1.00

The proposed method is also robust against rotation attacks. The center of gravity problem of Chicken is able to result in $corr^L_{avg} = 0.97$, however, the error-free watermark extraction is possible in the case of Cow and Dance in terms of both $corr^L$ and $corr^H$.

6.5 Conclusions

In this chapter, we proposed blind watermarking methods for 3-D mesh sequences. To be robust against various attacks such as spatial and temporal attacks, our methods use temporal wavelet transform. Two methods are introduced. The first method directly embeds the watermark into each axis of wavelet coefficients, and the second method into their vector norm. To increase the capacity of the watermark bits to be embedded, we divide the bins by using low frequency coefficients which are less sensitive to several attacks. As the vertices with the identical connectivity index over whole frames belong to one bin, their high frequency coefficients are also assigned into the same bin. For embedding a watermark, we modify the variance of wavelet coefficients in high (or middle) frequency band. The distribution of temporal wavelet coefficients can be approximated to Laplacian distribution which concentrates on zero mean. The high frequency coefficients in each bin can be divided by two groups according to the sign of the high frequency coefficient. To embed the watermark bit $+1$ (or -1), the variance of positive group is changed relatively bigger (or smaller) than negative group.

Two methods are introduced. Each method has focused on the capacity and invisibility, respectively. To increase the capacity of watermark, The first method modifies the variance of the distribution from temporal wavelet coefficients of each axis, independently. The other changes the second moment of the distribution of the temporal wavelet coefficients of vertex norms for better invisibility of the watermark and the robustness against rotation attacks.

From the modification method of high frequency coefficients of vertex norms, we propose the watermarking method by modifying the distribution of both low and high frequency coefficients. The method utilized our previous methods introduced in Chapter 5 and in Chapter 6.3.1. The method utilized variance shifting method in Chapter 5 to embed the watermark into each low frequency frames. The distribution of temporally high frequency coefficients of the vector norm is modified by the method in Chapter 6.3.1.

We showed that the proposed methods are fairly robust against various attacks that are probably concerned in copyright protection of 3-D mesh sequences. In particular, we also showed that the watermark in low frequency could be the complement of the previous methods in Chapter 6.3.1 to improve the capacity and the robustness. The extraction of the watermark which is embedded in each low frequency frame is also possible in spatial domain without temporal wavelet transform. This is one of the attractive properties of our method, however, the performance needs to be improved by considering the artifact by the high frequency coefficients and the filter-bank choice for temporal wavelet transform.

Our method does not require the original mesh sequence. The hidden watermark bit can be easily retrieved by simple comparison between the second moment of positive group and of negative group.

Chapter 7

General Conclusions and Perspectives

In this thesis, we have proposed wavelet transform based digital watermarking methods for 3-D surface meshes and mesh sequences for the application of copy-right protection.

The performance of the digital watermarking method is determined by the watermark embedding primitive and the hiding technique. Although wavelet transform needs more computational time for watermark embedding and extraction process, it provides several advantages in terms of invisibility, capacity and robustness against attacks.

In the first part of the thesis, we introduced two watermarking methods for 3-D triangular surface meshes based on multiresolution wavelet analysis.

In Chapter 4, we proposed a robust blind watermarking method for 3-D triangular surface meshes. In the previous works, [KANA98] embeds the watermark into the wavelet coefficient vector in a certain order, they have to use the original mesh to re-orient the topological synchronization of the vertices. An extended work [UCCH04] is introduced for the blind detection, but their methods are limited to the input mesh with regular subdivision connectivity as was in [KANA98]. By using irregular wavelet analysis scheme, our proposal can be applied to irregular as well as regular meshes without re-meshing. To recover the topological information after inverse wavelet transform, vertex re-ordering algorithm is used as a pre-processing for forward wavelet transform and a post-processing for inverse wavelet transform. The watermarks which are embedded into the wavelet coefficients are visually imperceptible while having the good robustness against geometrical attacks and distortion-less attacks. Through the simulation results, we proved that the vertex and face re-ordering procedure as pre-processing for both watermark embedding and extraction makes our method be robust against connectivity reordering attacks. We also showed that multiresolution embedding has more capacity while keeping the robustness against connectivity reordering as well as geometrical attacks. However, the watermark which is embedded into the wavelet coefficients in a certain order by spread spectrum method can not be retrieved after connectivity information is destroyed by topological attacks. For the robustness against topological attacks, pre-processing such as registration and re-sampling is required as in [PRAU99, YIN01], or the robustness against topological attacks cannot be guaranteed [KANA98, UCCH04, KIM05].

In Chapter 5, we developed a blind watermarking method which is robust against topological attacks. As previous wavelet analysis based 3-D mesh watermarking methods [KANA98, UCCH04, KIM05] embed the watermark information into wavelet coefficients arranged in a certain order, statistical features of scale coefficients on approximated meshes are used as a watermark carrier [KIM06b, KIM06a]. The statistical features might be less sensitive than the scale (or wavelet) coefficients themselves in a certain order. The proposed methods modify the distribution of scale coefficients to embed the watermark. As the number of scale coefficients decreases after wavelet analysis, our methods are more applicable to relatively large models. The watermark extraction is able to be done directly from vertex norms in spatial domain. We do not use wavelet analysis in the watermark detection. Then, it is not necessary to use any pre-processing such as registration, re-sampling and re-ordering for wavelet analysis in the process of watermark detection. Furthermore, the technique can reduce drastically the computational complexity and processing time. While this technique can slightly decrease the robustness against geometrical attacks, it allows our methods to be robust against topological attacks. Assuming that the distribution of each bin can be approximated uniform distribution, the mean or variance of the uniform distribution is modified by histogram mapping function. As the watermark detection process does not require wavelet analysis or any pre-processing, the embedded watermark can be simply calculate the mean or variance of each bin. Through simulations, we proved that the proposed has fairly good performance in terms of the watermark transparency and robustness against various attacks. Even though the proposed methods are not highly robust, our attempts demonstrate a possibility, blind watermarking based on wavelet analysis for 3-D mesh model.

In the second part of the thesis, we have presented digital watermarking methods for mesh sequence based 3-D animation using temporal wavelet transform.

In Chapter 6, we presented watermarking methods for 3-D mesh sequences. The main idea is to transform coordinates of vertex with the identical connectivity index along temporal axis using wavelet transform and to use the distribution of wavelet coefficients. The wavelet (high frequency) coefficients from temporal wavelet transform can be approximated to Laplacian probability density function with symmetric property. Then we modify the distribution of high frequency coefficients by histogram mapping function. We change the variance of high (or middle) frequency coefficients according to the watermark bit. Due to the use of the distribution, our method can retrieve the hidden watermark without any information about original mesh sequences in the process of watermark detection. All vertices are divided into groups, namely bins, using the distribution of low-frequency coefficients.

As the vertices with the identical connectivity index over whole frames belong to one bin, their high frequency coefficients are also assigned into the same bin. Then, the watermark is embedded into each axis of the high frequency coefficients or into the vector norms of the high frequency coefficients. The methods generates bins from the average frame of low frequency frames and modify the second moment of the probability distribution of high frequency coefficients in the bin. The robustness and the capacity of the methods could be increased by the number of total frames, and the number of vertices of the frame as well as the regularity of the vertices for computing the center of gravity. Through simulations we show that the proposed is fairly robust against various attacks that are probably concerned in copyright protection of 3-D mesh sequences.

In Chapter 6, we have also presented another watermarking method for 3-D mesh sequences. From the property of the wavelet decomposition, the method modifies both low and high frequency coefficients after temporal wavelet transform to increase the robustness and the capacity of the watermark to be embedded. For each low frequency frames, we modify the variance of the distribution of vertex norms by the method introduced in Chapter 5. We also modify the distribution of temporally high frequency coefficients by the second method in Chapter 6. We showed that the watermark in low frequency could be the complement of the first and second method to improve the capacity and the robustness. The extraction of the watermark which is embedded in each low frequency frame is also possible in spatial domain without temporal wavelet transform. This is one of attractive properties of our method, however, the performance needs to be improved by considering the artifact by the high frequency coefficients and the filter-bank choice for temporal wavelet transform.

Several directions for future work remain open. Since the proposed methods are evaluated on only a few meshes, it would be interesting to know the behavior of the proposed methods when they are applied to larger diversity of 3-D models as those available in very different areas, for example, CAD models, 3D laser scanning, terrain modeling, molecule design, 3D volume triangulation for numerical simulation, and so on. In addition, the robustness against clipping attacks needs be considered for further research. It would be improved by using the local distributions of each subpart of the original meshes.

Bibliography

[ALLI03] P. Alliez and C. Gotsman. Recent advances in compression of 3d meshes, 2003.

[ARNO00] Michael Arnold. Audio watermarking: Features, applications, and algorithms. In *IEEE International Conference on Multimedia and Expo (II)*, pages 1013–1016, 2000.

[ASPE02] N. Aspert, D. Santa-Cruz, and T. Ebrahimi. Mesh: Measuring errors between surfaces using the hausdorff distance. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, volume I, pages 705 – 708, 2002. `http://mesh.epfl.ch`.

[BEND96] W. Bender, D. Gruhl, N. Morimoto, and Aiguo Lu. Techniques for data hiding. *IBM Syst. J.*, vol. 35, no. 3–4, pp. 313–336, 1996.

[BENE99] Oliver Benedens. Geometry-based watermarking of 3d models. *IEEE Comput. Graph. Appl.*, vol. 19, no. 1, pp. 46–55, 1999.

[CAYR03] F. Cayre. *Contributions au tatouage des maillages surfacique 3D.* PhD thesis, ENST/TSI et UCL/TELE, 2003.

[CHO07] J. W. Cho, R. Prost, and H. Y. Jung. An oblivious watermarking for 3-d polygonal meshes using distribution of vertex norms. *IEEE Trans. Signal Processing*, vol. 55, no. 1, pp. 142–155, Jan 2007.

[CHO04] Jae-Won Cho, Min-Su Kim, Rémy Prost, Hyun-Yeol Chung, and Ho-Youl Jung. Robust watermarking on polygonal meshes using distribution of vertex norms. In Ingemar J. Cox, Ton Kalker, and Heung-Kyu Lee, editors, *IWDW*, volume 3304 of *Lecture Notes in Computer Science*, pages 283–293. Springer, 2004.

[CHO06] J.W. Cho, M.S. Kim, S. Valette, H.Y. Jung, and R. Prost. 3-d dynamic mesh compression using wavelet-based multiresolution analysis. In *Proceedings of ICIP 2006*, pages 529–532, october 2006.

[CIGN98] P. Cignoni, C. Rocchini, and R. Scopigno. Metro: Measuring error on simplified surfaces. *Computer Graphics Forum*, vol. 17, no. 2, pp. 167–174, 1998.

[COX97] I. Cox, J. Kilian, T. Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Trans. Image Processing*, vol. 6, no. 12, pp. 1673–1687, 1997.

[COX02] I. Cox, M. L. Miller, and J. A. Bloom. *Digital watermarking.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.

[CRAV97] Scott Craver, Nasir D. Memon, Boon-Lock Yeo, and Minerva M. Yeung. Can invisible watermarks resolve rightful ownerships? In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 310–321, 1997.

[DCI] Digital Cinema System Specification V1.0, Final Approval July 20, 2005, Digital Cinema Initiatives, LLC Member Representatives Committee. `http://www.dcimovies.com/DCI_Digital_Cinema_System_Spec_v1.pdf`.

[CIFE] CineFence, Royal Philips Electronics. `http://www.business-sites.philips.com`.

[FIEL88] D.A. Field. Laplacian smoothing and delaunay triangulation. *Communication and Applied Numerical Methods*, vol. 4, pp. 709–712, 1988.

[FREY99] P.J. Frey and P.L. George. *Maillages: application aux methodes d'elements finis.* Hermes, Paris, France, 1999.

[FRID01] J. Fridrich, M. Goljan, and R. Du. Invertible authentication. In *In Proceedings of the SPIE vol. 3971, Security and Watermarking of Multimedia Contents III*, pages 197–208, 2001.

[FRID02] J. Fridrich, M. Goljan, and R. Du. Lossless data embedding–new paradigm in digital watermarking. *EURASIP J. Appl. Signal. Process.*, no. 2, pp. 185–196, 2002.

[GARL97] M. Garland and P. S. Heckbert. Surface simplification using quadric error metrics. In *SIGGRAPH '97*, pages 209–216, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.

[GONZ92] R. Gonzalez and R. Woods. *Digital Image Processing*, pages 518–548. Addison-Wesley Publishing Company, 1992.

[GRUH96] Daniel Gruhl, Anthony Lu, and Walter Bender. Echo hiding. In *Information Hiding*, pages 293–315, 1996.

[GUSK04] Igor Guskov and Andrei Khodakovsky. Wavelet compression of parametrically coherent mesh sequences. In *SCA '04: Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 183–192, New York, NY, USA, 2004. ACM Press.

[HART98] F. Hartung and B. Girod. Watermarking of uncompressed and compressed video. *Signal Processing*, vol. 66, no. 3, pp. 283–301, 1998.

[HERO03] Herodotus. *The Histories (Penguin Classics).* Penguin Classics, ISBN 0140449086, April 2003.

[HIST] Michaele Stewart. People, Places Things: Histiaeus, Greek Mythology: From the Iliad to the Fall of the Last Tyrant. `http://messagenet.com/myths/ppt/Histiaeus_1.html`.

[HORM02] K. Hormann. An easy way of detecting subdivision connectivity in a triangle mesh. Technical Report 3, Department of Computer Science 9, University of Erlangen, May 2002.

[IBAR03] Lawrence Ibarria and Jarek Rossignac. Dynapack: space-time compression of the 3d animations of triangle meshes with fixed connectivity. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 126–135, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[KALI03] A. Kalivas, A. Tefas, and I. Pitas. Watermarking of 3d models using principal component analysis. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*, volume 5, pages 676–679, 2003.

[KANA98] S. Kanai, D. Date, and T. Kishinami. Digital watermarking for 3d polygon using multiresolution wavelet decomposition. In *Proc. Sixth IFIP WG 5.2 GEO-6*, pages 296–307, Tokyo, Japan, December 1998.

[KARN00] Zachi Karni and Craig Gotsman. Spectral compression of mesh geometry. In Kurt Akeley, editor, *Siggraph 2000, Computer Graphics Proceedings*, pages 279–286. ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2000.

[KEJA03] Arun Kejariwal. Watermarking. *IEEE Potentials*, Oct./Nov.:37–40, 2003.

[KIM02] Min-Su Kim. A Study on Digital Video Watermarking based on Discrete Wavelet Transform. Master's thesis, Department of Information & Communication Engneering, Yeungnam University, 2002.

[KIM05] M. S. Kim, S. Valette, H. Y. Jung, and R. Prost. Watermarking of 3d irregular meshes based on wavelet multiresolution analysis. *Lecture Notes in Computer Science*, vol. 3710, pp. 313–324, September 2005.

[KIM06a] M.S. Kim, J.W. Cho, R. Prost, and H.Y. Jung. Wavelet anaysis based blind watermarking for 3-d surface meshes. In *In Proc. of IWDW 2006*, volume 5, pages 477–480, November 2006.

[KIM06b] M.S. Kim, J.W.Cho, H.Y. Jung, and R. Prost. A robust blind watermarking for 3d meshes using distribution of scale coefficients in irregular wavelet analysis. In *Acoustics, Speech, and Signal Processing, 2006. Proceedings. (ICASSP '06). 2006 IEEE International Conference on*, volume 5, pages 477–480, May 2006.

[KIM06c] M.S. Kim, R. Prost, H.Y. Chung, and H.Y. Jung. A blind watermarking for 3-d dynamic mesh model using distribution of temporal wavelet coefficients. In *In Proc. of IWMRCS 2006*, volume 5, pages 477–480, September 2006.

[KIMT00] Tae-Hoon Kim, Jehee Lee, and Sung Yong Shin. Robust motion watermarking based on multiresolution analysis. *Comput. Graph. Forum*, 19(3):189–198, 2000.

[KOCH96] E. Koch, J. Rindfrey, and J. Zhao. Copyright protection for multimedia data. In *In Digital Media and Electronic Publishing*, pages 203–213, London, 1996. Academic Press.

[LANG96] G. C. Langelaar, R. L. Lagendijk, , and J. Biemond. Realtime labeling methods for mpeg compressed video. In *In Proc. 18th Symp. Information Theory in the Benelux*, pp 25–32, 1996.

[LANG97] Gerhard.C Langelaar, J.C.A van der Lubbe, Reginald L. Lagendijk. Robust Labeling Methods for Copy Protection of Images. *SPIE-EI'97 - Storage and Retrieval for Still Images and Video Databases*, Feb. 1997. Vol. V No. 3022, pp. 289–309, 1997.

[LEE03] Suk-Hwan Lee, Tae-Su Kim, Byung-Ju Kim, Seong-Geun Kwon, Ki-Ryong Kwon, and Kuhn-Il Lee. 3d polygonal meshes watermarking using normal vector distributions. In *Proc. of ICME*, pages 105–108, 2003.

[LEE05] Seung-Woo Lee; A. Senot, Ho-Youl Jung and R. Prost. Regularized marching cubes mesh. *Image Processing, 2005. ICIP 2005. IEEE International Conference on.* vol.3, pp. 788–791, Sept. 2005.

[LENG99] Jerome Edward Lengyel. Compression of time-dependent geometry. In *SI3D '99: Proceedings of the 1999 symposium on Interactive 3D graphics*, pages 89–95, New York, NY, USA, 1999. ACM Press.

[LIYI99] Ying Li and Xinbo Gao and Hongbing Ji. A 3D Wavelet Based Spatial-Temporal Approach for Video Watermarking. In *ICCIMA '03: Proceedings of the 5th International Conference on Computational Intelligence and Multimedia Applications*, pages 260–266, Washington, DC, USA, 2003.

[LOOP87] Charles T. Loop. Smooth subdivision surfaces based on triangles. Master's thesis, Department of Mathematics, University of Utah, 1987.

[LORE87] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169. ACM Press, 1987.

[LOUN94] M. Lounsbery. *Multiresolution Analysis for Surfaces of Arbitrary Topological Type*. PhD thesis, Dept. of Computer Science and Engineering, U. of Washington, 1994.

[MINT97] Fred Mintzer, Jeffrey Lotspiech and Norishige Morimoto. Safeguarding Digital Library Contents and Users: Digital Watermarking. Corporation for National Research Initiatives, 1997.

[OHBU98] R Ohbuchi, H Masuada, and M Aono. Watermarking three-dimensional polygonal models through geometric and topological modifications. *IEEE Journal on Special Areas in Communications*, vol. 16, no. 4, pp. 551–560, 1998.

[OHBU97] R. Ohbuchi, H. Masuda, and M. Aono. Watermarking three-dimensional polygonal models. In *Proceedings of the fifth ACM international conference on Multimedia*, pages 261–272. ACM Press, 1997.

[OHBU01] Ryutarou Ohbuchi, Shigeo Takahashi, Takahiko Miyazawa, and Akio Mukaiyama. Watermarking 3d polygonal meshes in the mesh spectral domain. In *GRIN'01: No description on Graphics interface 2001*, pages 9–17, Toronto, Ont., Canada, 2001. Canadian Information Processing Society.

[PAYA02] F. Payan and M. Antonini. Multiresolution 3d mesh compression. In *IEEE Int. Conf. on Image Processing ICIP'02*, volume 2, pp. 245–248, September 2002.

[PAYA05] F. Payan and M. Antonini. Wavelet-based compression of 3d mesh sequences. In *Proceedings of IEEE ACIDCA-ICMI'2005*, Tozeur, Tunisia, november 2005.

[PETI99] Fabien A. P. Petitcolas, Ross J. Anderson, and Markus G. Kuhn. Information hiding — A survey. *Proceedings of the IEEE*, vol. 87, no. 7 pp. 1062–1078, 1999.

[PITA98] I. Pitas. A method for watermark casting on digital image *Circuits and Systems for Video Technology, IEEE Transactions on* , vol.8, no.6, pp.775–780, Oct 1998.

[PODI98] C.I. Podilchuk and Wenjun Zeng. Image–adaptive watermarking using visual models. *Selected Areas in Communications, IEEE Journal on*, 16(4):525–539, 1998.

[PODI01] C.I. Podilchuk and Wenjun Zeng. Digital watermarking: Algorithms and applications. *IEEE Signal Processing Magazine*, vol. 18, no. 4, pp. 33–46, 2001.

[PODI97] C.I. Podilchuk and Wenjun Zeng. Perceptual watermarking of still images. In *In Electron. Proc. IEEE SPS Workshop Multimedia Signal Processing, Princeton, NJ*, pp. 363–368, June 1997.

[PRAU99] E. Praun, H. Hoppe, and A. Finkelstein. Robust mesh watermarking. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pp. 49–56, 1999.

[QUIS95] Jean-Jacques Quisquater, O. Bruyndonckx and B. Macq. Spatial method for copyright labeling of digital images. *Proc. IEEE Workshop on Nonlinear Signal and image processing*, pp. 456–459, 1995.

[SCHR02] W. Schroeder, K. Martin, and B. Lorensen. *The Visualisation ToolKit*. Kitware, 2002.

[SEO01] Yong-Seok Seo, Min-Su Kim, Ha-Joong Park, Ho-Youl Jung, Hyun-Yeol Chung, Young Huh, and Jae-Duck Lee. A secure watermarking for jpeg-2000. In *In Proc. of IEEE International Conference on Image Processing 2001 (ICIP 2001)*, vol. 2, pp. 530–533, October 2001.

[SPAN66] Edwin Henry Spanier. *Algebraic topology*. McGraw-Hill series in higher mathematics. McGraw-Hill, New York [u.a.], 1966.

[SURA03] Vitaly Surazhsky, Pierre Alliez, and Craig Gotsman. Isotropic remeshing of surfaces: a local parameterization approach. In *Proceedings of 12th International Meshing Roundtable*, pp. 215–224, 2003.

[SWAN98a] Mitchell D. Swanson, Bin Zhu, and Ahmed H. Tewfik. Multiresolution scene-based video watermarking using perceptual models. *IEEE Journal on Special Areas in Communications*, 16(4):540–550, 1998.

[SWAN98b] Mitchell D. Swanson, Bin Zhu, Ahmed H. Tewfik, and Laurence Boney. Robust audio watermarking using perceptual masking. *Signal Processing*, 66(3):337–355, 1998.

[SWEL95] W. Sweldens. The lifting scheme: A new philosophy in biorthogonal wavelet constructions. In A. F. Laine and M. Unser, editors, *Wavelet Applications in Signal and Image Processing III*, pages 68–79. Proc. SPIE 2569, 1995.

[TIRK93] A. Z. Tirkel, G. A. Rankin, R. M. van Schyndel, W. J. Ho, N. R. A. Mee, and C. F. Osborne. Electronic watermark. In *Digital Image Computing, Technology and Applications (DICTA'93)*, pages 666–673, Macquarie University, Sidney, 1993.

[TOUM98] Costa Touma and Craig Gotsman. Triangle mesh compression. In *Graphics Interface*, pages 26–34, 1998.

[UCCH04] F. Uccheddu, M. Corsini, and M. Barni. Wavelet-based blind watermarking of 3d models. In *Proceedings of the 2004 multimedia and security workshop on Multimedia and security*, pages 143–154. ACM Press, 2004.

[VALE04a] S. Valette and R. Prost. Multiresolution analysis of irregular surface meshes. *IEEE Trans. Visual. Comput. Graphics*, vol. 10, no. 2, pp. 113–122, Mar-Apr 2004.

[VALE04b] S. Valette and R. Prost. A wavelet-based progressive compression scheme for triangle meshes: Wavemesh. *IEEE Trans. Visual. Comput. Graphics*, vol. 10, no. 2, pp. 123–129, Mar-Apr 2004.

[VALE03] S. Valette, J. Rossignac, and R. Prost. An efficient subdivision inversion for wavemesh-based progressive compression of 3d triangle meshes. In *IEEE Int. Conf. on Image Processing ICIP'03*, volume 1, pages 777–780, Barcelona, Spain, September 14-17 2003.

[VANS94] R. G. van Schyndel, A. Z. Tirkel, and C. F. Osborne. A digital watermark. In *International Conference on Image Processing*, volume 2, pages 86–90, Austin, Texas, U.S.A., 1994.

[VORS01] Jens Vorsatz, Christian Rössl, Leif Kobbelt, and Hans-Peter Seidel. Feature sensitive remeshing. *Comput. Graph. Forum*, vol. 20, no. 3, 2001.

[WATS94] Andrew B. Watson. Perceptual optimization of DCT color quantization matrices. In *Image Processing, 1994. ICIP 1994. IEEE International Conference on*, pages 100–104, 1994.

[WATS96] Andrew B. Watson, Gloria Y. Yang, Joshua A. Solomon, and John Villasenor. Visual thresholds for wavelet quantization error. In B. Rogowitz and J. Allebach, editors, *Human Vision and Electronic Imaging*, volume 2657, pages 382–392, 1996.

[WONG03] Peter H. W. Wong, Oscar C. Au, and Y. M. Yeung. Novel blind multiple watermarking technique for images. *IEEE Trans. Circuits Syst. Video Techn.*, vol. 13, no. 8, pp. 813–830, 2003.

[WU05] Hao-Tian Wu and Yiu-ming Cheung. A Reversible Data Hiding Approach to Mesh Authentication. *WI '05: Proceedings of the The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)*, pp. 774–777, 2005.

[YAMA04] Shuntaro Yamazaki. Watermarking motion data. In *Proc. Pacific Rim Workshop on Digital Steganography (STEG04)*, pages 177–185, november 2004.

[YIN01] K. Yin, Z. Pan, J. Shi, and D. Zhang. Robust mesh watermarking based on multiresolution processing. *Computers and Graphics*, vol. 25, no. 3, pp. 409–420, June 2001.

[YU03a] Zhiqiang Yu, Horace H. S. Ip, and L. F. Kwok. A robust watermarking scheme for 3d triangular mesh models. *Pattern Recognition*, vol. 36, no. 11, pp. 2603–2614, 2003.

[YU03b] Zhiqiang Yu, Horace H. S. Ip, and L. F. Kwok. Robust watermarking of 3d polygonal models based on vertice scrambling. In *Computer Graphics International*, pages 254–257. IEEE Computer Society, 2003.

[ZAFE05] Stefanos Zafeiriou. Blind robust watermarking schemes for copyright protection of 3d mesh objects. *IEEE Trans. Visual. Comput. Graphics*, vol. 11, no. 5, pp. 596–607, 2005.

List of Personal Bibliography

[Kim et al., 2006d], Kim, M.S, Cho, J.W., Prost, R., Jung, H.Y., A Blind Watermarking for 3-D Mesh Sequences Using Temporal Wavelet Transform of Vertex Norms, Journal of Korean Institute of Communication and Sciences, submitted.

[Kim et al., 2006c], Kim, M.S, Prost, R., Chung, H.Y., Jung, H.Y., A Blind Watermarking for 3-D Dynamic Mesh Model Using Distribution of Temporal Wavelet Coefficients, in *IWMRCS 2006's Proceedings*, LNCS 4105, Istanbul, Turkey, 2006, (pp. 257-264).

[Kim et al., 2006b] Kim, M.S, Cho, J.W., Jung, H.Y., Prost, R., A Robust Blind Watermarking for 3D Meshes Using Distribution of Scale Coefficients in Irregular Wavelet Analysis, in *ICASSP 2006's Proceedings*, volume 5, IEEE, Toulouse, France, 2006, (pp. 477-480).

[Kim et al., 2006a], Kim, M.S, Cho, J.W., Prost, R., Chung, H.Y., Jung, H.Y., Wavelet Analysis Based Blind Watermarking for 3-D Surface Meshes, in *IWDW 2006's Proceedings*, LNCS 4283, Jeju, Korea, 2006, (pp. 123-137).

[Kim et al., 2005], Kim, M.S, Valette, S., Jung H.Y., Prost, R., Watermarking of 3D Irregular Meshes Based on Wavelet Multiresolution Analysis, in *IWDW 2005's Proceedings*, LNCS 3710, Siena, Italy, 2005, (pp. 313-324).

FOLIO ADMINISTRATIF

THESE SOUTENANCE DEVANT L'INSTITUT NATIONAL DES SCIENCES APPLIQUEES DE LYON

Nom : KIM **Date de soutenance :** 26 février 2007
Prénom : Min-Su

Titre : Wavelet Transform based Digital Watermarking for 3-D Surfaces Meshes and Mesh Sequences

Nature : Doctorat **Numéro d'ordre :** 2007 ISAL 00000
Formation doctorale : Sciences de l'Information, des Dispositifs et des Systèmes
Filière : Instrumentation, Système, Signal & Image

Code B.I.U. Lyon : T 50/210/19 / et bis **Classe :**

Abstract: This thesis deals with digital watermarking methods for copyright protection of 3-D surface meshes and mesh sequences. The proposed methods are based on wavelet analysis of the geometry of the surface meshes, or on wavelet analysis of the vertex coordinate signals along the time axis.

3-D static meshes. The first proposal embeds the watermark into the L2 norm of the geometric wavelet coefficients (vectors) by using spread spectrum method. The Cartesian coordinates should be converted into the spherical coordinates. The method can directly process semi-regular or irregular meshes. For the invariance of the wavelet coefficients after inverse wavelet transform, it is necessary to synchronize the connectivity by re-ordering the vertex indexes from a reference vertex. This approach ensures the blind detection which does not require the original meshes during watermark extraction procedures. The second proposal is based on a statistical approach. The histogram of the L2 norm of scale coefficients (approximation mesh) was divided into regular bins. The mean (or the variance) of each bin is modified according to the watermark by a non-linear exponential transformation. The watermark can be extracted from the distribution of the scale coefficients after wavelet analysis, as well as from the distribution of vertex norms of the watermarked meshes without using the wavelet analysis. This proposal is more robust against topological attacks than the first one.

3-D mesh sequence (3-D dynamic meshes with fixed connectivity, 3D+t) The first proposal is based on the vertex coordinate signals along the time axis, and the second uses the vertex norm signals along the time axis of the mesh sequence. As a result, the number of signals to decompose by wavelet analysis equals three times of the number of vertices for the first proposal and equals the number of vertices for the second. The first proposal embeds the watermark into the temporal wavelet (high frequency) coefficients of each coordinates of the sequence. In this proposal, the histogram of L2 norm of the scale (low frequency) coefficients is divided into distinct bins with equal range according to their magnitude. Since the distribution of temporal wavelet coefficients can be approximated to Laplacian distribution, the variance of the distribution of each bin is asymmetrically modified according to the watermark bits to be embedded. This method is robust against frame-averaging attack and frame-by-frame attacks. In the second proposal, the Cartesian coordinates of the vertices are converted into the spherical coordinates and the L2 norm of each vertex is wavelet transformed along the time axis. Similar to the first proposal, the variance of the distribution of wavelet coefficients is modified according to the watermark. This proposal which uses L2 norm outperforms the first proposal in terms of the invisibility of the watermark and the robust against rotation attacks. Finally, we propose the method which embeds the watermark into both wavelet and scale coefficients. This method allows extracting the watermark from the distribution of scale and wavelet coefficients of each frame. In addition, it is possible to extract the watermark from a single frame in spatial domain. Note that all of our proposals do not require the mesh sequence (or original meshes) in the procedures of watermark extraction. The robustness against geometrical attacks (additive noise on the vertex coordinates, uniform vertex coordinates quantization, low-pass filtering, cropping, rotation, translation, scaling, frame-dropping for mesh sequence), and topological attacks (random vertex index reordering, simplification, subdivision) are evaluated on the semi-regular and irregular meshes.

Mots-Clés : 3-D Surface Meshes, 3-D Mesh Sequence, Digital Watermarking, Copyright Protection, Wavelet, Blind Detection

Laboratoire de recherche :
Centre de Recherche Et d'Applications en Traitement des Images et du Signal (CREATIS), UMR CNRS 5220, U630 Inserm

Directeurs de thèse : Cotutelle INSA-Lyon /Yeungnam University Korea : Rémy PROST et Ho-Youl JUNG

Président de jury :
Composition du jury : Francis SCHMITT (rapporteur), Ki-Ryong, KWON (rapporteur), Isabelle MAGNIN, Kook-Yeol, YOO, Rémy PROST (directeur), Ho-Youl JUNG (directeur)