A STATISTICAL WATERMARKING SCHEME FOR 3D MESH MODELS BASED ON VERTEX SMOOTHNESS MEASURE

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

MASTER OF TECHNOLOGY
IN
SIGNAL PROCESSING AND DIGITAL DESIGN

Submitted by:

NEHA SHARMA

(2K17/SPD/09)

Under the supervision of

PROF. J. PANDA

DEPARTMENT OF ELECTRONICS AND COMMUNICATION
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

JUNE, 2019

DEPARTMENT OF ELECTRONICS AND COMMUNICATION
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I, Neha Sharma, 2K17/SPD/09, of M.Tech, hereby declare that the project Dissertation Titled "A Statistical watermarking scheme for 3D mesh models based on vertex smoothness measure" which is submitted by me to the Department of Electronics and Communication, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **NEHA SHARMA**

Date:

DEPARTMENT OF ELECTRONICS AND COMMUNICATION
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Project Dissertation titled "A Statistical watermarking scheme for 3D mesh models based on vertex smoothness measure" which is submitted by Neha Sharma, Roll No-2K17/SPD/09, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the students under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi

Date:

DR. J. PANDA
Professor
Department of Electronics and communication
Delhi Technological University, Delhi
SUPERVISOR

ACKNOWLEDGEMENT

I thank GOD almighty for guiding me throughout the semester. I would like to thank all those who have contributed to the completion of my project and helped me with valuable suggestions for improvement.

I am extremely grateful to Prof. J. Panda, Department of Electronics and Communication, for providing me with best facilities and atmosphere for the creative work guidance and encouragement.

Above all I would like to thank my parents without whose blessings, I would not have been able to accomplish my goal.

……

NEHA SHARMA

ABSTRACT

3D mesh watermarking has sparked interest, thanks to the sudden increase of graphic content and animated motion movies, however the state of analysis of watermarking 3D models continues to be in its infancy as compared to research work in image and video watermarking. 3D Watermarking provides a deterrent to piracy of 3D models by embedding a hidden piece of data within the original content. Watermarking algorithms have a basic demand that the watermark ought to be imperceptible to avoid being detected and not cause visible distortion to the viewer. The watermark ought to even be sturdy to face up to unintentional attacks. It's conjointly desired that the watermark insertion capability ought to be as high as attainable to face up to intentional attacks and to permit insertion of multiple or redundant or bio-metric watermarks. Insertion of high density imperceptible watermark can create it very tough for an attacker to seek out the watermark and so create substantial changes within the 3D model to get rid of or change the watermark. However, inserting huge amounts of data as watermark will cause distortion; therefore designing of watermarking algorithms involves a trade-off between imperceptibility, capacity and robustness.

This thesis discusses blind watermarking of 3D models within the spatial domain. Although it's been notable that oblivious (or blind) watermarking schemes are less sturdy and robust than non oblivious ones, they're helpful for numerous applications wherever a host signal isn't out there within the watermark detection procedure. The target of this research work is to explore innovative ways in which to insert the most quantity of secret info into 3D mesh models while not inflicting palpable distortion and conjointly create it tough for the attacker to guess where the watermark was inserted and therefore the quantity of watermark inserted.

The proposed methodology estimates the local smoothness variation of the mesh to pick vertices for inserting a watermark. Smoothness variation of the surface represented by the 1- ring neighborhood of every vertex is computed by the average angle difference between

the surface normal and the average normal. Every vertex of the mesh is labeled in one of the bins corresponding to variable degrees of local smoothness variation from the low to the moderate to the highest variation. Vertices with the label of moderate local smoothness variation are then picked for the insertion of a random watermark. The projected methodology employs a statistical approach to embed watermark on selected vertices that changed the distribution of vertex norms consistent with the watermark bit to be embedded. Histogram mapping functions are introduced to modify the distribution as per the watermark requirement. These mapping functions are devised to lessen the visibility of watermark as much as possible. Since the statistical features of vertex norms are invariant to the distortion-less attacks, the proposed method is sturdy against rotation, translation and scaling. Moreover, our method use an oblivious watermark detection scheme, which may extract the watermark while not requiring the cover mesh model. Simulation results prove that the inserted watermark is invariant against affine operations, noise and smoothing attacks, and at the same time imperceptibility is retained.

The thesis is structured as follows:

Chapter-1 gives an introduction about watermarking, its categories, requirements and applications, also giving an overview of the trade-off involved in designing desired watermarking schemes.

Chapter-2 gives a background of 3D mesh entities and basic terminologies used in thesis

Chapter-3 gives a comprehensive literature review of the contemporary 3D watermarking algorithms.

Chapter-4 describes proposed watermarking methods in detail, including their insertion and embedding algorithm based on statistical features and vertex smoothness measure of a 3D mesh.

Chapter-5 shows the experimental results and analysis of the proposed method against various attacks**.**

Chapter-6 gives the conclusion and the future work of the thesis.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

-

LIST OF SYMBOLS AND ABBREVIATIONS

1.1- 3D- Three Dimensional

4.1- α- Strength Factor

4.2- K_n- Transformation parameter

5.1- VSNR- Vertex Signal to Noise Ratio

5.1- RMSE- Root Mean Square Error

5.2- Corr- Correlation

CHAPTER 1 INTRODUCTION

Recent years have seen an ascent in the accessibility of digital multimedia system content. Today, digital media's documents can be distributed via the world Wide net to an incredible number of individuals without much effort and money. Moreover, in contrast to traditional analog duplication, with which the standard of the duplicated content is degraded, digital tools will simply manufacture great amount of excellent copies of digital documents in an exceedingly short amount of time. This ease of digital multimedia distribution over the web, beside the likelihood of unlimited duplication of this data, threatens the intellectual property (IP) rights more than ever. Thus, content owners are thirstily seeking technologies that promise to safeguard their rights.

Cryptography is perhaps the most common technique for shielding digital content since it has a well-established theoretical basis and developed terribly with success as a science. The content is encrypted before delivery and a key is provided to the legitimate owner (who has bought it). However, the seller is unable to find how the product is handled when it is decrypted by the client. Encryption protects the content throughout the transmission solely. Once transmitted to the receiver, data should be decrypted so as to be valuable. Once decrypted, the information isn't any longer protected and it becomes vulnerable. The client could end up to be a pirate distributing prohibited copies of the decrypted (unprotected) content. Therefore, encryption should be complemented with a technology that may still shield the precious knowledge even when it's decrypted. This is often the purpose where watermarking comes in.

The digital media have been widely used to produce several digital products, for instance, individuals can acquire, duplicate, process, and distribute the digital media comparatively easily by several of the prevailing tools and also the web. As a result, these facilities also are exploited by pirates who use them illicitly for his or her personal gains to violate the legal rights of the digital content providers. The digital watermarking has been introduced as a good complementary to the traditional encryption for the digital watermark

may well be embedded into the various sorts of digital media, as well as pictures, audio data, video data, and three-dimensional graphical models like 3D polygonal models.

Digital watermarking is a technique designed to hide information in a certain kind of digital data. Embedded watermarks are often used to enforce copyright, data authentication or to add info to the data. Ideally, the watermark shouldn't interfere with the intended purposes of the data. A watermarking technique has 2 stages: watermark embedding and detection respectively. Most of the analysis on watermarking has focused on watermarking audio data, still pictures, or video, whereas audio data consists of one-dimensional time variable signals, pictures are 2-D mappings of digital data distributed on an oblong lattice. When applied on still pictures, the watermarking algorithms can be classified in those which are embedded in the space domain or in a transform domain like DCT.

Many three-dimensional (3D) objects are currently described in 3D meshes to actually replicate the topological structures of the objects. Among numerous illustration tools, triangular meshes give an efficient means to represent 3D mesh models. With high demand and recognition of 3D models and considering the price, time, and energy needed to make such models comes the menace of widespread amerceable repetition of 3D models. Regardless of the growing curiosity in digital watermarking of multimedia system data, watermarking of 3D geometrical models has received more or less little or no attention by the research community. one of the pivotal reasons is that geometric knowledge is per se advanced to handle, in addition loads of numerous attacks will be thought of that don't seem to be attainable within the 2D and 1D cases, therefore it's highly troublesome to develop strong watermarking algorithms for 3D models. Today the applications using and managing 3D geometry data are quickly increasing in numbers, for instance, for the image and video case loads of process tools to de-noise, compress, transmit, enhance, analyze and edit these sorts of signals exist. The arrival of recent tools to process geometric data is extremely useful for 3D watermarking technology. In addition to this, 3D watermarking sets a novel set of issues that weren't present in the image and video cases; geometric data has intrinsic curvature, topology and no implicit ordering (with relation to the regular sampling of an image): it's not a straightforward 2D to 3D

extension. Moreover a 3D model may undergo a lot of advanced and complex attacks with relation to image and video media type, therefore it's troublesome to increase the well-consolidated image and video watermarking algorithms to this new sort of media. The consequence is that while loads of techniques and strategies to introduce copyright info in image and video are developed and tested with sensible performances, solely few algorithms to hide confidential information (for IPR, authentication and so on) into a 3D model are developed in order to beat the above issues most of the systems proposed to date exploit the information of the original, non marked, mesh for watermark recovery. The practical utility of non-blind schemes is limited due to need of original mesh model at the extraction side, thus the necessity to develop new blind schemes for 3D watermarking.

1.1. 3d Watermarking Categories

In digital watermarking, a digital code, or watermark, is inserted into the 3D model, referred to as the host or cover model, in order that a given piece of information is indissolubly tied to that. This info can later be used to prove possession, determine a misappropriating person, trace the model dissemination through the network, or just inform users regarding the rights-holder or the permitted uses. The way watermarking algorithm recover the watermark from the model has a strong impact on practical applications; it's then common to classify digital watermarking techniques by their decoding processes.

1.1.1. Blind vs. non blind.

A watermarking algorithm is blind if it doesn't need to compare the marked and unmarked documents to recover the watermark. Conversely, a watermarking algorithm isn't blind if it wants the initial data to extract the information from the marked document. Blind techniques are generally referred to as oblivious.

1.1.2. Readable vs detectable.

In this case we tend to distinguish between algorithms that introduce a code that can be read while not knowing it beforehand, and those that insert a mark that may solely be detected, that is, a user can solely verify whether or not a given code is contained within

the document. Detectable watermarking is sometimes referred to as 1-bit watermarking as a result of the detector output which is simply affirmative or no.

1.1.3. Robust vs fragile

A robust methodology aims to find the embedded message even after the object suffered from a serious level of attacks. These classes of methods are usually designed for the aim of copyright protection. On the opposite hand, a fragile message ought to disappear all once any attack happens to the 3D mesh model. A decent fragile watermarking algorithm ought to be able to locate the region being changed. Fragile watermarking is employed for the mesh authentication and tamper detection.

As per the requirements, a watermarking system must encompass the most important properties that are robustness, i.e. the ability to survive manipulations, unobtrusiveness, and capacity.

1.2. Requirements of 3D watermarking

1.2.1. *Watermark capacity*

Although generally the watermark capacity doesn't rely upon the particular algorithm, however it's associated with the characteristics of the host signal, of the allowed embedding distortion and of the attack strength. We refer to the capacity of a given technique as the quantity of information bits that the watermark is in a position to convey. in this sense, capacity is a elementary property of any watermarking algorithm, that fairly often determines whether or not a method will be productively employed in a given context or not. Usually speaking, capacity requirements continually struggle against two other vital requirements, that's imperceptibility and robustness. Having aforementioned this, it's obvious that the capacity of any 3D watermarking system is in relation with the complexity of the given mesh, where by mesh complexity we intend the number of faces and vertices it contains. Thus, a mesh with numerous faces can convey a lot of bits than a simple mesh with a few faces.

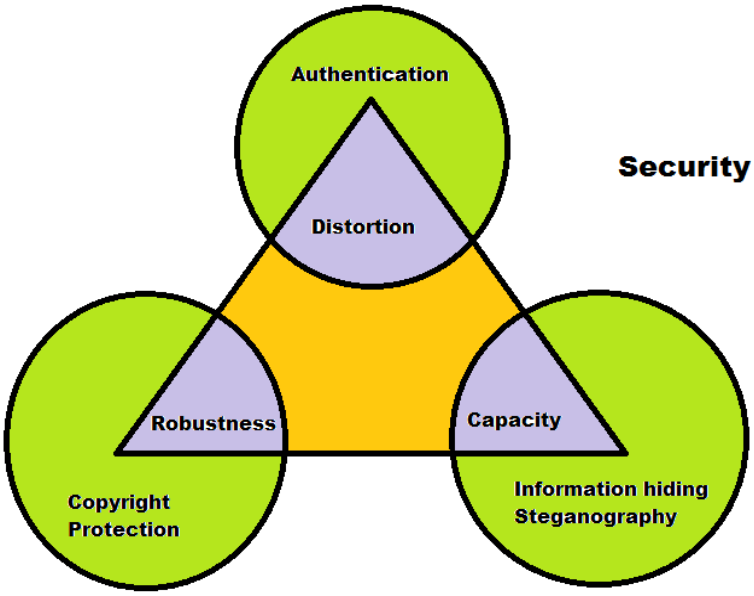


Fig.1.1: Property triangle of 3D watermarking

1.2.2. *Imperceptibility*

The watermarked model should maintain identical visual quality as that of the original one. The importance of this property depends on the intended use of the model; usually the intended use is viewing; therefore the watermarked model and the original one should seem similar in visual scrutiny. This can be a vital point because often a user sees a 3D model in an interactive way. On the contrary, images and video don't permit such an in depth user-interaction, therefore it's so far easier to cover the watermark using appropriate perceptual masks. It's necessary to underline that for a few applications the imperceptibility of the watermark might not be an adequate requirement. This is often the case, for instance, when we need to research the deformations of cultural heritage goods by periodic 3D acquisition of their surfaces.

1.2.3. *Robustness*

Every watermarking algorithm to be employed in IPR (Intellectual Property Rights) applications needs to be robust against manipulations, usually known as attacks, of the watermarked media. The matter with 3D watermarking is that a great deal of attacks is possible

Figure 1.1 illustrates the relation among the 3 factors: robustness, capacity and distortion with the security taken into consideration. Attempting to enhance any one part might limit the effectiveness of the others, as an example, we might acquire a better robustness if we relax the necessity of the distortion and embed fewer bits; however the trade-off is that the object is also terribly distorted from the first one. Increasing the capacity implies that one bit of message will be carried by fewer vertexes. Thus, this might scale back the robustness of the watermarking algorithm. Security suggests that how much is the chance that the embedded message will be recovered and removed by malicious users. It's relatively not as vital as the other 3 factors in watermarking methods however it can't be neglected. How to find a correct balance among these aspects is the most difficult issue in the analysis of 3D watermarking algorithms.

1.3. Embedding domain

The first step towards the definition of watermarking algorithms consists in the selection of the host features, i.e. the choice of a group of properties of the cover 3D model that may bear the watermark info. Of course, several possibilities exist here, however, in this thesis; we tend to focus solely on geometric and topological features.

1.3.1. Geometric Features.

The main geometric features of a mesh are its vertices. One potential way to insert the watermark is to change the position or the normals of vertices (vertex normals are associated with the curvature of the mesh). both these entities are altered by perturbing the coordinates of mesh vertices.

1.3.2. Topological Features.

These features are associated with the connectivity of the mesh vertices. Usually, a collection of connected vertices is chosen by using geometric features. Then, the topology of those vertices is redefined to encode one or additional bits. In the proposed approach we've decided to use as embedding features, the vertices position because vertices contain most of the data of the mesh.

1.4. Applications of 3D Watermarking

There are several potential applications of 3D watermarking due to 2 reasons. Firstly, any info may be embedded into the object. Secondly, the watermarked models can be used as original ones, because watermarking methods aim to insert the message while not modifying the appearance of the 3D model. The content of the embedded info can be employed in numerous ways in which, the easiest application is to shield the 3D object. Copyright info, like author or creation date etc, can be embedded so as to shield the intellectual property of the 3D model. In a virtual 3D object market, an artist creates some extraordinary 3D models. Then he can save the data such as web site, price, even barcode to the object. Once the author finds some unauthorized copy over the web, he can claim his copyright by retrieving his own watermark code from the 3D model. Fragile watermarking can be used as an authentication or tampering detection tools. The watermark will disappear once a watermarked object is changed and ideally the detected info can tell where and how the mesh is changed. In a 3D database, we are able to incorporate the data like, database index, mesh description, category etc, into the mesh. It'd be terribly cumbersome to copy a 3D scene if it contains thousands of objects of varied varieties and sizes. However if each object is embedded with a message describing its location within the scene, it'd create the rendering sort of a puzzle game. This will save ample space for storing and rendering time.

CHAPTER 2 BACKGROUND

Watermarking techniques insert invisible data into the multimedia content. The covertly embedded data is termed the watermark and may accommodates a users unique ID, cryptographic keys, copyright ownership messages, access conditions of the content, logos, image, biometrics, or content-based info. The watermark embedding and retrieval method is assisted by a secret key, within which lies info on where and to what extent has the initial content been changed so as to accommodate the watermark. Imperceptibility is a strong demand of every watermarking scheme, because the watermark mustn't distort the original media or interfere with its intended use or function. Robustness is important to assure that common signal processing, geometric operations and malicious modifications don't impact the detection or retrieval of the watermark. The motive is to facilitate content owners to prove their possession by retrieving the watermark from a pirated media so litigate against the offender.

Features are extracted in the spatial domain. The spatial domain watermarking schemes are usually less robust to attacks like compression and noise addition. They however survive cropping attack and are less complex. They insert watermark in spatial domain by either modifying vertex positions or modifying the connectivity of the vertices.

2.1. Terminology

Before we enter into the details, it is necessary to understand the specific terminology used throughout this thesis.

Cover medium

An original digital medium (3D object) without being watermarked or processed is called cover medium (or cover object).

Stego medium

When the cover medium is watermarked by some watermarking algorithms, watermarked medium object is then referred to as stego medium.

Watermark

Watermark is the message being embedded to the cover object. We use w to represent the message to be embedded, while wˆ denotes the retrieved message from the object that is watermarked or attacked.

Watermark embedding

It is the process of embedding the watermark into the cover object.

Watermark detection

It is the process of extracting the embedded message.

Robustness

We measure the robustness of a watermarking algorithm using the Bit Error Rate, i.e. the ratio between the correctly detected bits and the total number of embedded bits.

Distortion

It means the similarity between the watermarked object and the original one.

Fig.2.1: A 3D mesh model

2.2. A 3D Mesh

A 3D mesh consists of 3 combinative entities: vertices, faces, and also the edges connecting the vertices. A vertex list provides the coordinates in 3D space of each and every vertex within the model and a face list that describes how the vertices are connected to each other. An edge list may be derived by traversing the face and vertex list. Figure 1 is an example of a wireframe mesh.

From another viewpoint, a mesh can even be fully represented by 2 types of info: the geometry information describes the 3D positions (coordinates) of all its vertices, while the connectivity info provides the adjacency relations between the different components. Mathematically, a 3D polygonal mesh containing M vertices and N edges can be modeled as a signal Z = {G, C}, where

$$V = \{v_i\}_{i=1,2,...,M} , v_i = (x_i , y_i , z_i) \quad (1)$$

$$C = \{(v_{k1} , v_{k2})\} , 1 \leq k1 \leq M, 1 \leq k2 \leq M, k1 \neq k2 \quad (2)$$

Each vertex element v_i in V is numbered by an index i and is described by its three-dimensional coordinates (x_i , y_i , z_i); C has N elements and each element stands for an edge connecting two different vertices indexed by k1 and k2, respectively.

The degree of a facet is the number of its component edges, and the valence of a vertex is defined as the number of its incident edges.

With the increasing capability of capturing, processing and visualizing 3D data, the intellectual property protection of 3D meshes has garnered lots of attention. Naturally, as a promising technique, watermarking seems to be a good candidate for solving this increasing problem. Fragile watermarks may also be used to authenticate the origin and integrity of the received 3D mesh data at the user end. Attacks on watermarked meshes play a vital role in the design of appropriate watermarking algorithms as they're way more intractable than their counterparts on images.

2.3. Vertex Smoothness Measure

Curved surface comprise of a number of smaller triangles to give the perception of smooth surface. Normal variation usually offers a good indication of the surface curvature. Gaussian and mean curvatures are the most commonly used measures for finding the curvature of a surface. However, these curvature measures capture the global characteristics of a surface. In this work we tend to use angle variation between surface normals and the average normal corresponding to a vertex to determine the vertex smoothness measure. Such computed measure is takes into account the local geometry of the surface. This measure is then used to determine the quantity of watermark to be added. The feature vector is a set of angles derived by computing the orientation of the surface normal's to the average normal of the triangular faces that form a 1-ring neighborhood for a vertex. This feature vector represents the curvature of the 1-ring vertex neighborhood. The length of the feature vector is equal to the valence of the vertex, which is the count of how many other vertices the vertex is connected to within the 3D model.

CHAPTER 3 LITERATURE SURVEY

Since three-D mesh watermarking techniques were introduced in [4], there are many attempts to boost the performance in terms of transparency and robustness. Ohbuchi et al. [4] proposed 3 watermarking schemes: triangle similarity quadruple, tetrahedral volume ratio (TVR), and a visible mesh watermarking technique. These schemes can be considered oblivious (or blind) techniques that can extract the watermark without reference to a cover mesh model; however they're not sufficiently robust against numerous attacks. For instance, TVR is extremely susceptible to remeshing, simplification, and rearrangement attacks. Beneden [5] proposed a watermark embedding technique that modifies the local distribution of vertex directions from the center point of model. The method is robust against simplification attack because the local distribution isn't sensitive to such operations. An extended scheme was additionally introduced in [6] to overcome a weakness to cropping attack. However, the method still needs preprocessing for reorientation throughout the process of watermark detection, because the local distribution basically varies with the degree of rotation. Yu et al. [7] proposed a vertex norm modification technique that perturbs the distance between the vertices to the center of model according to the watermark bit to be embedded. It employs, before the modification, scrambling of vertices for the aim of preserving the visual quality. Note that it's not an oblivious technique and additionally needs preprocessing like registration and resampling. Some multi-resolution based methods have additionally been introduced [8]–[10]. Kanai et al. [10] proposed a watermarking algorithm based on wavelet transform. Similar approaches, using Burt–Adelson style pyramid and mesh spectral analysis, were additionally published in [9] and [10], respectively. The multi-resolution techniques might accomplish a high transparency of watermark however haven't been used as an oblivious scheme since the connectivity info of vertices should be precisely best-known for multi-resolution analysis in the watermark extraction method. Recently, there are some trials that apply the spectral analysis based techniques on to point-sampled geometry that's independent of vertex connectivity info [11], [12]. However, they're not oblivious schemes. Although it's been noted that oblivious schemes are less robust than non

oblivious ones, they're more helpful for numerous applications wherever a host signal isn't accessible in the watermark detection procedure. For instance, owner identification and copy control systems cannot refer to original information [13]–[14]. moreover, the utilization of non oblivious watermarking will cause one to confuse the proof of ownership if an unlawful user asserts that he's the copyright holder with a corrupt watermarked information as his original [15], [16]. In this thesis, our interests focus on developing an oblivious watermarking.

Three-dimensional polygonal mesh models have serious difficulties for watermark embedding. Where image information are represented by brightness (or amplitudes of RGB elements in the case of color images) of pixels sampled over a regular grid in 2 dimension, three-D polygonal models don't have any distinctive representation, i.e., no implicit order and connectivity of vertices [7]. This creates synchronization problem throughout the watermark extraction that makes it tough to develop robust watermarking techniques. For this reason, most techniques developed for other kinds of transmission aren't effective for three-D meshes. Furthermore, a range of complicated geometrical and topological operations might disturb the watermark extraction for assertion of ownership.

The geometrical attacks include adding noise, smoothing, and so on. Vertex rearrangement, simplification, and re-meshing constitute the class of topological attacks. These attacks are often reclassified into 2 categories: distortion and distortion-less attacks [19]. Distortion attacks include adding noise, simplification, smoothing, re-meshing, clipping, and so on, which can cause visual deformation of the stego mesh model. Most typical watermarking techniques of three-D polygonal mesh models are developed to be robust primarily against the distortion attacks [5], [6], [8]–[10], [18]. On the opposite hand, distortion-less attacks include similarity transforms and vertex rearrangement. These attacks are successfully overcome by some non-oblivious watermarking strategies [17], [9], [18]. However, they might be more serious attacks to oblivious watermarking as they could fatally destroy the hidden watermark without any perceptual changes of stego mesh model. Clearly, it's required to develop an oblivious watermarking technique that's robust against distortion-less as well as distortion attacks.

The recent algorithms on 3D watermarking are classified to 2 types: spatial-domain techniques and frequency-domain techniques [20]. The first kind inserts the watermark by

directly changing the mesh geometry or connectivity, whereas the second kind inserts the watermark by changing the frequency domain coefficients after mesh transformation. The insertion method in the first kind is usually easier and quicker than in the second kind, the inserted watermark is less impalpable and robust for the operations of 3D meshes. However, the watermarking method in the second sort is greater complicated and slower than the first kind because of the need to transform and reverse transform.

According to the extraction operation, watermarking approaches are classified into blind and non-blind extraction. A blind extraction watermark needs solely the secret key in the extraction stage and doesn't need the original model, however non blind extraction watermark needs the secret key and original model in the extraction stage [21,22].

This thesis focuses on inserting big quantity of hidden information into the original models by dividing vertices into several bins based on their feature vector while not occasioning visible deformation and additionally it's difficult for the attacker to speculate locations of the watermark insertion.

The method proposed in this thesis takes inspiration from [1,2,3,24] by separating vertices into bins and using info about normals to faces. This can be performed by presenting the algorithm that utilizes the unsupervised mechanism. in this work, binning is used to categorize mesh vertices into suitable and unsuitable selections for being watermark carrier based on the feature vector. Then watermark is embedded in the suitable vertices that come after binning based on vertex norm,

We additionally apply a statistical approach that modifies the distribution of vertex norms to hide watermark info into host three-D meshes as in [1]. In contrast with [5], we tend to modify the distribution of vertex norms rather than normal distribution to hide watermark info similar to [5], we tend to split the distribution into distinct sets referred to as bins and insert one bit per bin. The distribution of vertex norms is changed by shifting the mean value of the distribution according to the water mark bit to be embedded. An identical approach has been used to shift the mean value in [29], where a constant is added to vertex norms. In particular, histogram mapping functions are used for the aim of elaborate modification. Since the statistical features are invariant to distortion-less attacks and less sensitive to numerous distortion ones with local geometric alterations, robustness

of watermark is simply achieved. In addition, the proposed methods use an oblivious watermark detection scheme

Gaussian and mean curvatures are the most commonly used measures for determining the curvature of a surface. A variety of curvature computing techniques are mentioned in [23]. However, these curvature measures capture the global characteristics of a surface. In this thesis local curvature variation is estimated to pick areas for watermark insertion. The computed curvature estimate is local and relative to the geometry of the surface. A positive Gaussian curvature value suggests that the surface is locally either a peak or a valley. A negative value suggests that the surface locally has saddle points. A zero value suggests that the surface is flat in a minimum of one direction (i.e., both a plane and a cylinder have zero Gaussian curvature). Since the calculation of low and high curvature is relative to the model, the value of the curvature estimate isn't significant. However, the variation in curvature is critical because it's used for choice of regions for embedding the watermark. Such regions are better qualified candidates for insertion of an imperceptible watermark as opposed to making choice based on globally computed Gaussian and mean curvature estimates.

CHAPTER 4 PROPOSED METHOD

4.1. Watermark Insertion

4.1.1. Preprocessing

Normalization of 3D models as a pre-processing step before insertion of a watermark makes the watermark resilient to modifications within the 3D model because of affine and scaling transformations. The center of mass of the 3D model is shifted to the origin and the model is scaled to fit in a unit cube

4.1.2. Local Geometry Representation

It is necessary that a good watermarking algorithm mustn't use connectivity information since mesh vertex reorder would simply destroy such a watermark. There's a requirement to outline a feature vector to represent the local geometry of the mesh to insert the watermark. These feature vectors would be then determined over a surface which represent the local geometry. In this project, 1-ring has been chosen to represent the local geometry of a 3D model. In future work, other local geometric structures like Voronoi rings, fixed radius 3D balls, and voxels may be used to represent local geometry and feature vectors derived based on these geometric structures may be exploited to classify the vertices for selection of insertion of watermark and optimize the quantity of watermark to be inserted. The 1-ring neighborhood of a vertex V is defined as the surfaces formed by that vertex V with its neighbors as shown in Figure. 4.1

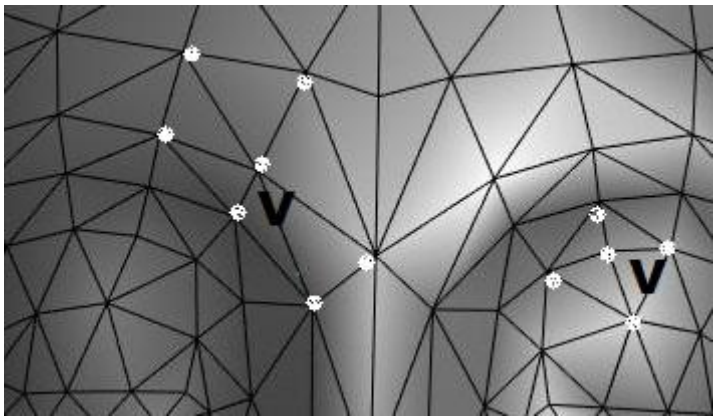


Fig..4.1: Vertices with degree-6 and degree-5 in 1-ring structure

3D triangular mesh models are represented by a set of vertices and a list of triangular faces formed by the vertices. A vertex v_i is a neighbor of another vertex v_j if an edge exists that connects v_i and v_j. The set of all the neighbors of a vertex v_i is called 1-ring of the vertex.

4.1.3. Feature Extraction

The basis of the use of these bio-inspired algorithms is based on the grounds of two key observations mentioned below.

The perception of distortion caused by inserting watermark is influenced by the surrounding geometry of the vertices, for instance, perturbation of an isolated vertex in 3D wouldn't cause the eye to perceive any modification in location of the vertex. However, if the isolated vertex is within the backdrop of a flat surface, the modification is more visible. If the same vertex is on a rough surface, the modification is impalpable. Thus, the absolute location of the vertex isn't necessary for watermarking; however its location relative to the local geometry determines whether or not the vertex is a smart or bad candidate for watermark insertion. If info is inserted in a specific vertex, it's going to be perceivable as distortion, whereas if info is inserted within the neighbourhood of the vertices along with the vertex under consideration, the distortion may be cloaked by the supporting geometry. The neighbourhood of the vertices ought to also be considered in the selection process to determine their suitability or ”fitness” for selection.

Therefore, based on the above key observations, feature vectors are defined which quantitatively model these observations. Based on the kind of local surface geometry, two features are defined.

Curvature is the amount by which a geometrical surface deviates from being flat. curved surface contains a number of smaller triangles to convey the perception of smooth surface as compared to what's required for a flat surface. Normal variation usually provides a decent indication of the surface curvature. for instance, if the surface is flat, all the surface normals are parallel to every other and there's zero deviation of the average normal from each of the surface normals. If the surface is smooth, the deviation of the surface normal from the average normal is in line with the deviation of the other surface normals from the

average normal. If the surface has uneven curvature, the deviation of the surface normals from the average normal may well be erratic.

Bumpiness in the wavelet domain is calculated by dividing the scalar coefficient of a low resolution model with the length of vector between its adjacent scalar coefficients.

Selection and determination of set of vertices as watermark carrier is considered the initial step in the proposed algorithm. It can be performed by binning 3D mesh vertices into suitable and unsuitable watermark carrier based on the feature vector. Feature vector is a group of angles calculated between the normals and the average normal of the polygonal faces which constitute a 1-ring for a vertex.

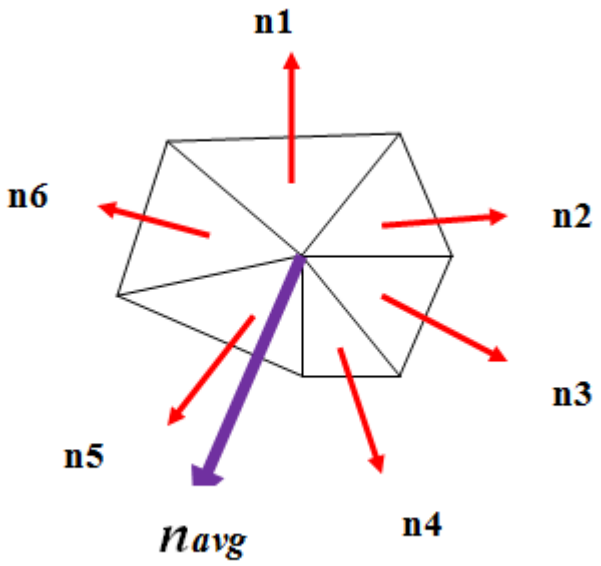


Fig. 4.2: Normals (n1; n2; …; n6) and average normal (n_{avr}) for a 1-ring vertex of degree 6

The procedure for determining the feature vector for vertices is:

Step 1 Compute normals nj to all face that is constituted by V and its adjacent vertices

Step 2 Determine the average n-avr of all the previous normals transiting over the vertex V in its one ring neighborhood.

Step 3 Compute the angles between the surface normal n_j and average normal n_{avr},

$$\text{Feature vector F} = (\vartheta_1, \vartheta_2, \vartheta_3, \vartheta_4, \ldots \ldots \ldots) \quad (3)$$

$$\vartheta_i = \cos^{-1}\left(\frac{n_i . n_{avg}}{|n_i||n_{avg}|}\right) \quad (4)$$

In order to perform clustering, feature vector is calculated of all vertices. It determines the topical geometry of an area. If the area is peak, the measure of angles will be high but in flat areas the measure of angles will be low. Peak and flat areas are ignored to achieve high transparency and consider their vertices as unsuitable watermark carriers. We only take into account the vertices of moderate value to be carriers of the watermark, by utilizing binning to group all mesh vertices into 8 groups according to the values of vertices ranging from low, then moderate up to high. The sequence of computing feature vector for 3D mesh model is depicted in Figure. 4.2

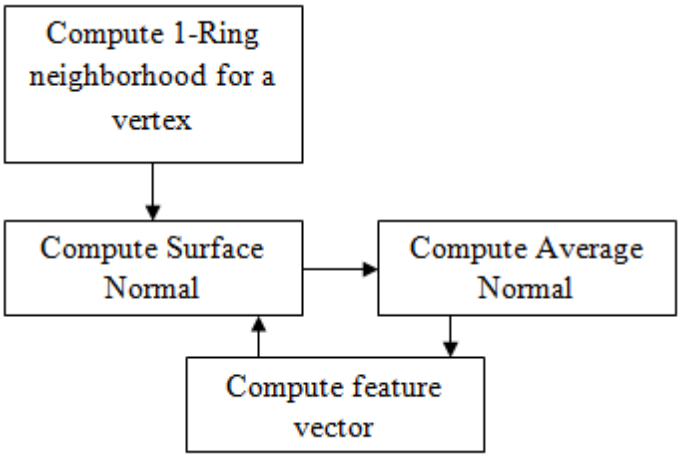


Fig. 4.3: Block diagram to calculate feature vector for a 3D mesh model

4.1.4. Watermark Insertion

Once the locations of suitable watermark carriers that is vertices are determined after binning process, a random watermark sequence w is required to be embedded in those locations, the embedding procedure is based on inserting watermark into the 3-D mesh model by changing the distribution of vertex norms. The distribution is divided into distinct sections, called as bins, each of which is used as a watermark embedding unit to

insert 1 bit of watermark. The embedding is performed by modifying the mean value of vertex norms greater or smaller than a reference value according to watermark bit that we want to embed. Suppose that the vertex norms of cover meshes are mapped into the interval [0,1] and have a uniform distribution over the interval. To insert a watermark bit of +1, the distribution is modified so that its mean value is greater than a reference value. 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 value.

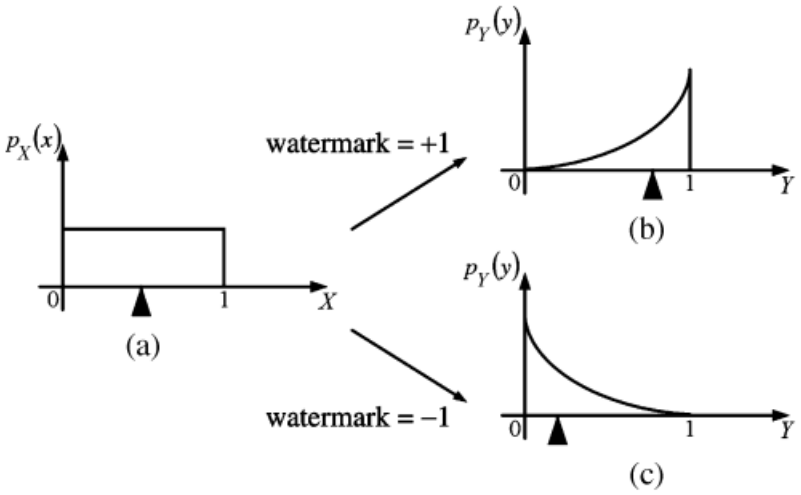


Fig. 4.4: Watermarking technique by shifting the mean of the distribution.

Firstly, Cartesian coordinates of a vertex on the cover mesh model are converted into spherical coordinates using-

$$r_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2 + \left(z_i - z_c\right)^2} \tag{5}$$

$$\theta_i = \tan^{-1} \frac{(y_i - y_c)^2}{(x_i - x_c)^2} \tag{6}$$

$$\varphi_i = \cos^{-1} \frac{(z_i - z_c)}{\sqrt{((x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2)}} \quad (7)$$

where L is the number of the vertex, is the center of gravity of the mesh model, and is the i-th vertex norm. The vertex norm is defined as the distance between each vertex and the center of gravity. The proposed method uses only vertex norms for watermarking and keeps the other two components untouched. Secondly, vertex norms are divided into N distinct bins with equal range, according to their magnitude. Each bin is used independently to hide single bit of watermark. If every bin is utilized for watermark embedding, we can embed at maximum N bits of watermark. To categorize the vertex norms into bins, maximum and minimum vertex norms are calculated beforehand.

The n-th B_n bin is defined as follows:

$$B_n = \left\{ r_{n,j} \middle| r_{min} + \frac{r_{max} - r_{min}}{N} . n < r_i < r_{min} + \frac{r_{max} - r_{min}}{N} . (n+1) \right\} \quad (8)$$

$$for\ 0 \le n \le N - 1, 0 \le i \le L - 1, 0 \le j \le M_n - 1$$

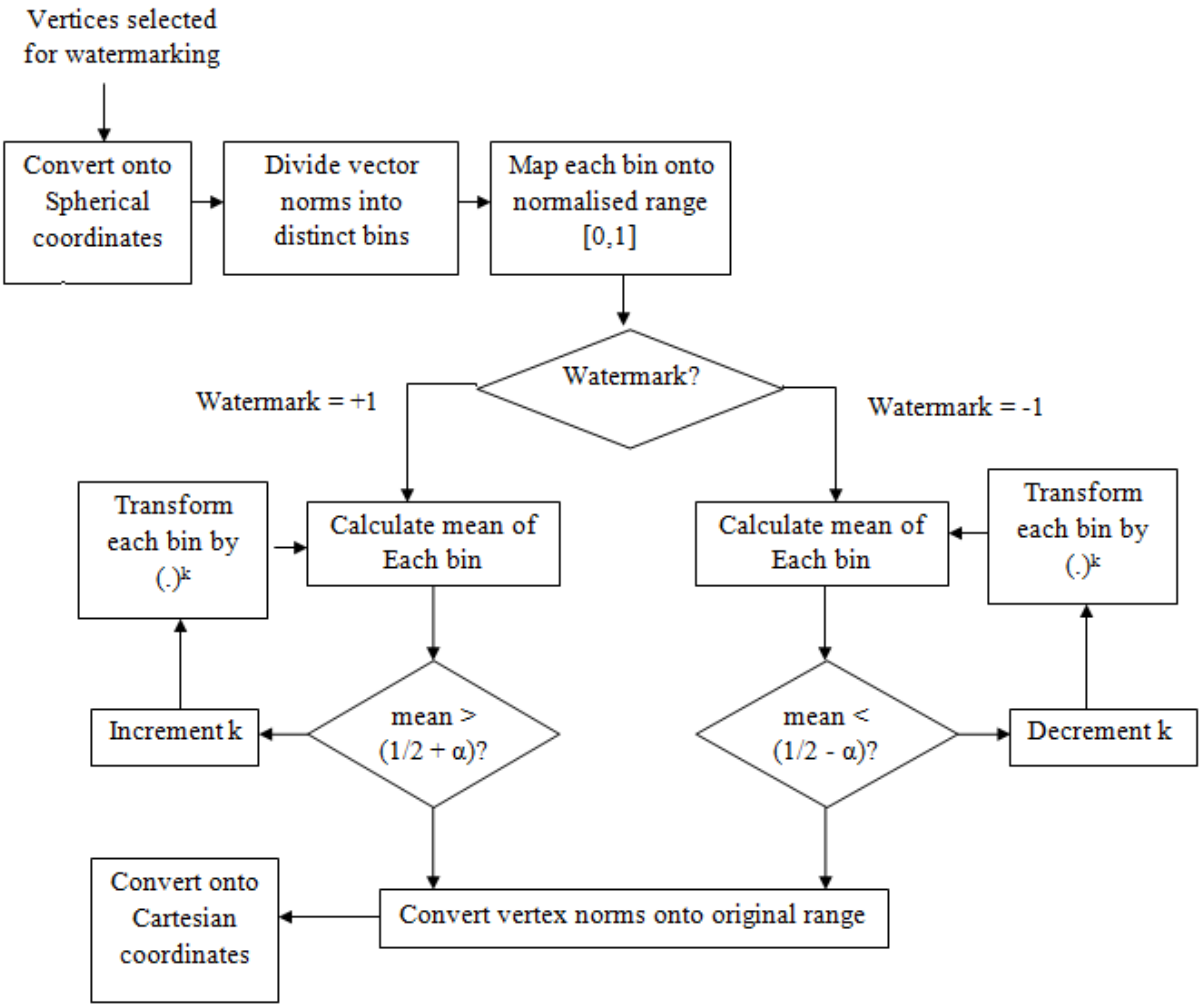


Fig. 4.5: Block diagram of watermark insertion process after selection of vertices

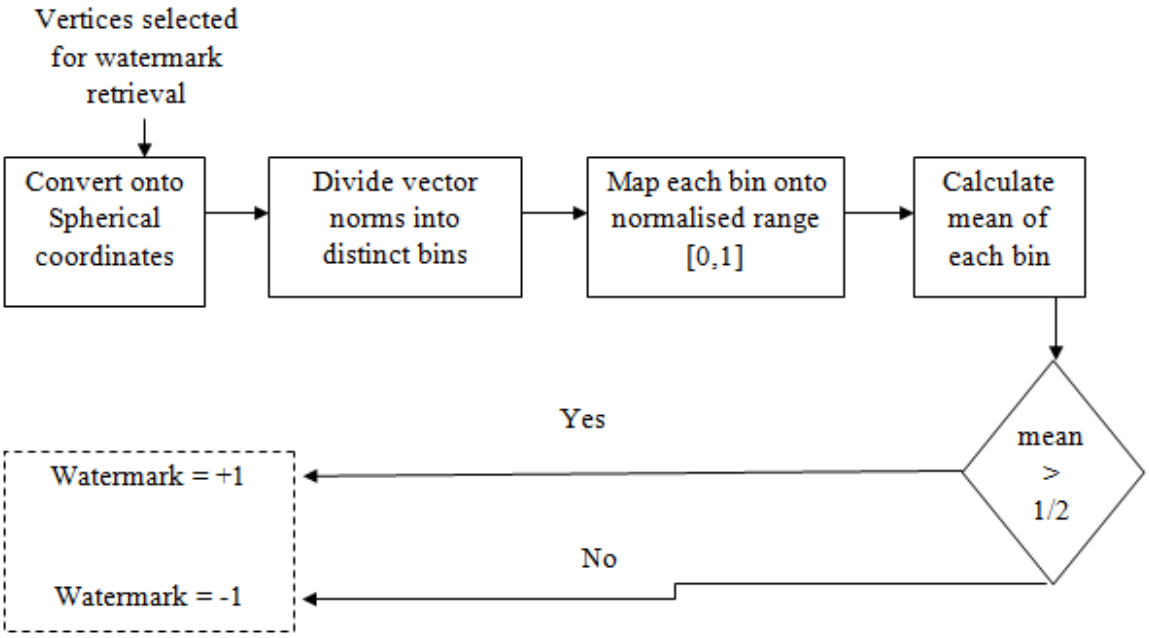


Fig. 4.6: Block Diagram of Watermark retrieval process after selection of vertices

Where M_n is the number of vertex norms belonging to the nth bin and r(i,j) is the j-th vertex norm of the nth bin. Thirdly, vertex norms belonging to the nth bin are mapped into the normalized range of [0,1] by-

$$\hat{r}_{n,j} = \frac{r_{n,j} - min_{r_{n,j \in B_n}}\{r_{n,j}\}}{max_{r_{n,j \in B_n}}\{r_{n,j}\} - min_{r_{n,j \in B_n}}\{r_{n,j}\}} \quad (9)$$

In our method, vertex norms in each bin are changed to shift the mean value. It is very necessary to assure that the transformed vertex norms also exist within the range of each bin. Or else, vertex norms belonging to a certain bin may shift into neighbor bins, which could have a grave consequence on the watermark extraction. We now use a histogram mapping function as proposed in [1], which can translate the mean to the desired level through modifying the value of vertex norms while staying within the proper range. The use of a mapping function is inspired from the histogram equalization techniques often used in image enhancement processing. For a given continuous random variable $\boldsymbol{X}$, the mapping function is defined as-

$$Y = X^k \; for \; 0 < k < \infty \; and \; k \in \boldsymbol{R} \quad (10)$$

where *Y is* the transformed variable and the parameter is a real value for 0<k<∞

Fig. 4.5 shows curves of the mapping function Y for different values of k. When the parameter k is chosen in the range [1 ∞), input variables are mapped into relatively small values. Therefore, increase in k leads to decrease in the value of the transformed variable. It means the reduction of mean value. On the other hand, the mean value increases for decreasing k when k lies in (0 1). Expectation of output random variable E[y] is represented as-

$$E[Y] = E[X^k] = \int_0^1 x^k p_x(x)dx = \frac{1}{k+1} \tag{11}$$

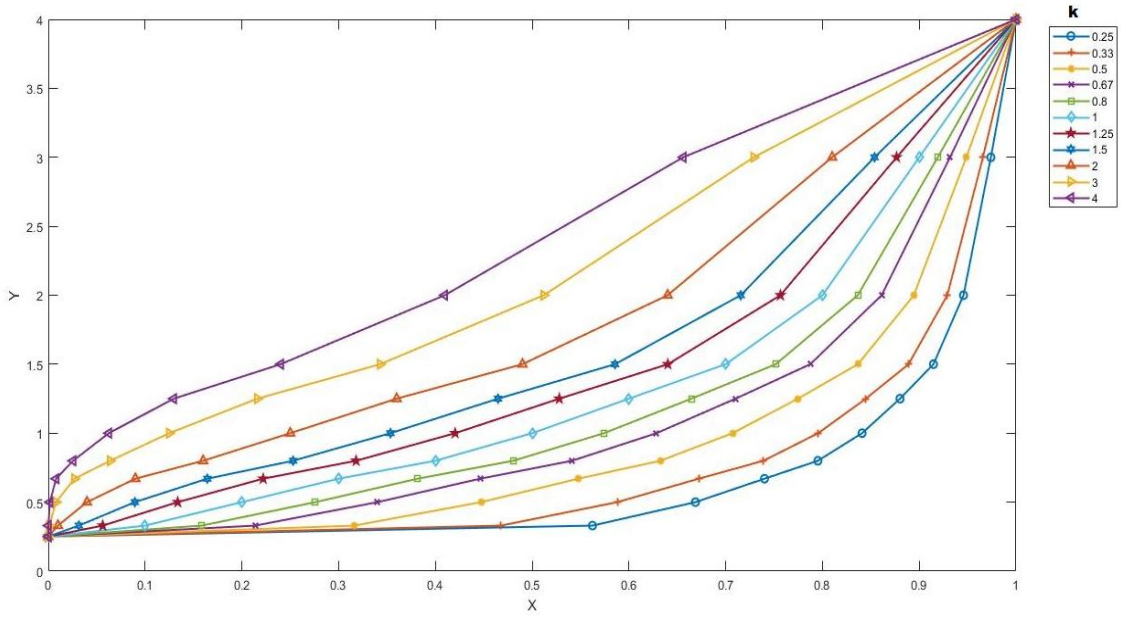


Fig. 4.7: Expectation of the output random variable via histogram mapping function

Fig. 4.7 shows the expectation value of the output of the mapping function over k. The expectation value decreases monotonically with the parameter k. Therefore, we can easily modify the mean value of the distribution by choosing a proper parameter. In particular, the mapping function not only assures to transform the variable within the limited range but also permits translation of the mean value to a desired level.

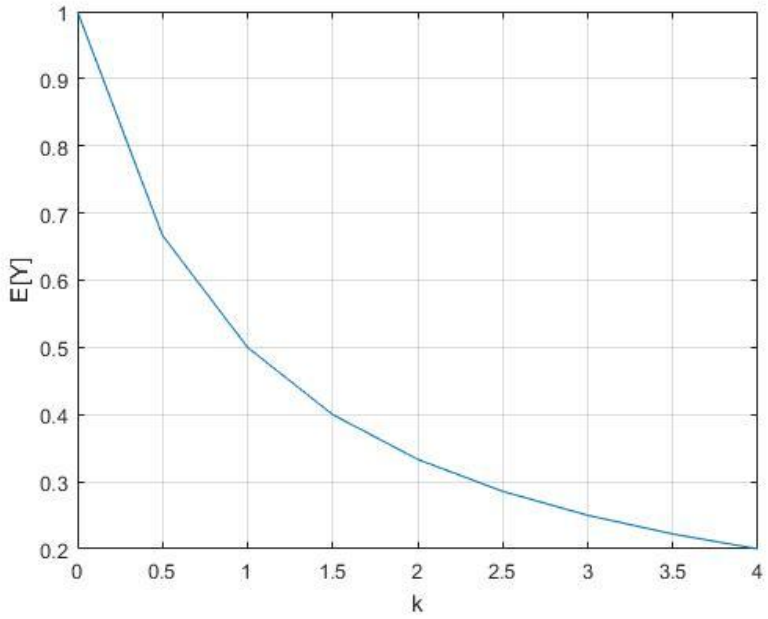


Fig. 4.8: Expectation of the output random variable via histogram mapping function with different values of k, assuming that the input random variable is uniformly distributed over unit range [0,1].

The fourth step of the proposed watermark embedding is to translate the mean value of each bin using vertex norms via the histogram mapping function. To insert a watermark bit mean of each bin is modifies as-

$$\widetilde{m_n} = \begin{cases} \frac{1}{2} + \alpha \, , if \; w_n = +1 \\ \frac{1}{2} - \alpha \, , if \; w_n = -1 \end{cases} \tag{12}$$

where α is the strength factor that can control the robustness and the transparency of watermark. The exact parameter K_n can be found directly from-

$$k_n = \begin{cases} \frac{1-2\alpha}{1+2\alpha} \; if \; w_n = +1 \\ \frac{1-2\alpha}{1+2\alpha} \; if \; w_n = +1 \end{cases} \tag{13}$$

The real vertex norm distribution in each bin is neither continuous nor uniform so k is experimentally determined parameter considering the tradeoff between the processing time and the precision error.

The fifth step is inverse computation of the third step. Modified vertex norms of each bin are mapped onto the original range using-

$$r_{n,J} = \acute{\widetilde{r_{n,J}}} \cdot \left(\underset{r_{n,j \in B_n}}{max} \{r_{n,j}\} - \underset{r_{n,j \in B_n}}{min} \{r_{n,j}\} \right) \tag{14}$$

In the end, the watermark insertion process is completed by combining all the bins and converting the spherical coordinates back to Cartesian coordinates.

A stego mesh model consisting of vertices represented in Cartesian coordinate is obtained using the-

$$\acute{x}_i = \acute{r}_i \cos\theta_i \ \sin\varphi_i + x_c \tag{15}$$

$$\acute{y}_i = \acute{r}_i \sin\theta_i \sin\varphi_i + y_c \tag{16}$$

$$\acute{z}_i = \acute{r}_i \cos\varphi_i + z_c \tag{17}$$

4.2. Watermark Extraction

Similar to insertion process, the stego mesh model is first converted to spherical coordinates. After finding the maximum and minimum vertex norms, the vertex norms are categorised into bins and mapped onto the normalized range of [0,1]. Then, the mean of each bin is computed and compared to the reference value of 1/2. The watermark hidden in the nth bin is retrieved using-

$$w_n = \begin{cases} +1, if\ \acute{m}_n > \frac{1}{2} \\ -1, \ if\ \ \acute{m}_n < \frac{1}{2} \end{cases} \tag{18}$$

The watermark extraction process does not need the original mesh, since it is a blind method.

CHAPTER 5 EXPERIMENTAL RESULTS AND ANALYSIS

The experiments were realized on Intel(R) Core(TM) i3-3217U and the proposed algorithm is executed in MATLAB R2017a and has been executed on several 3D mesh models sourced from The Stanford 3D Scanning Repository hosted by the Stanford University and Large Geometric Models Archive hosted by the Georgia Institute of Technology as shown in Fig 5.1

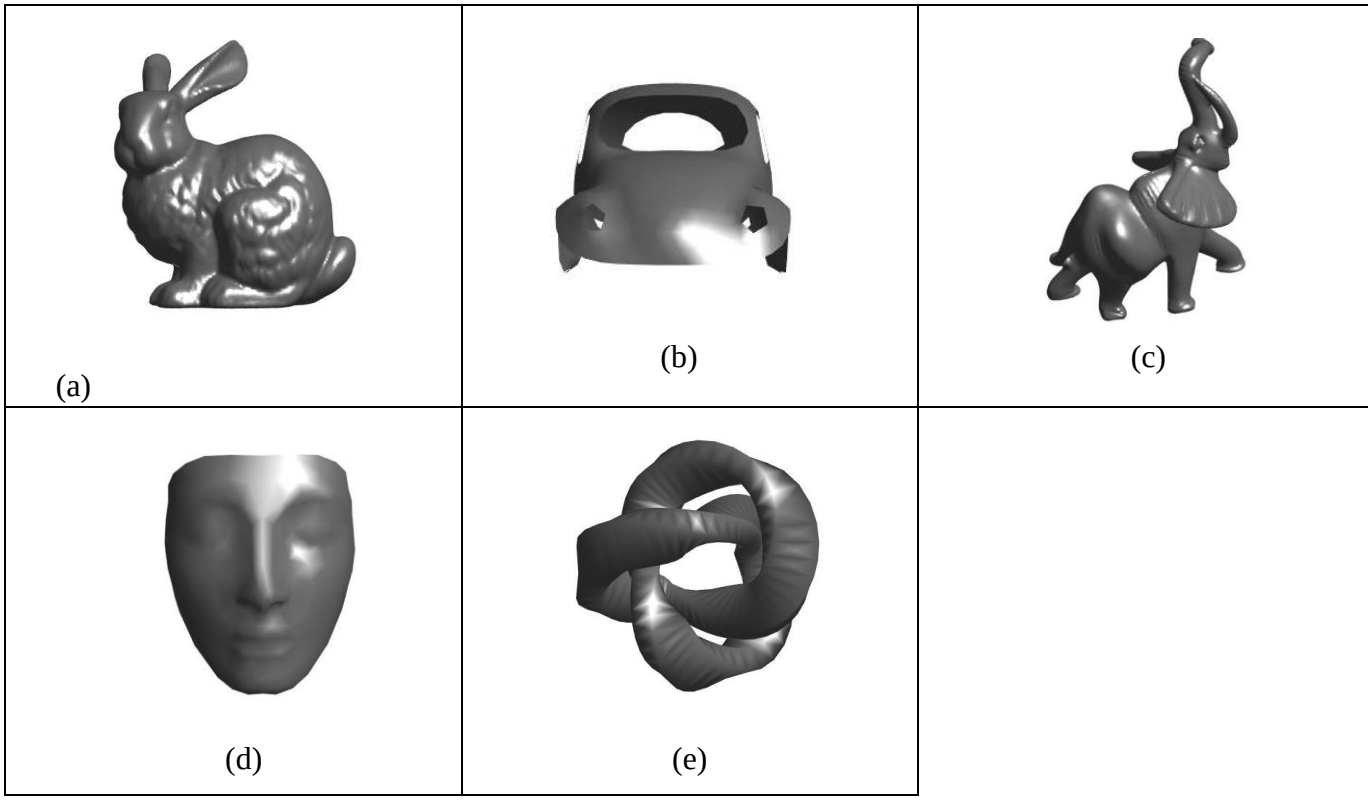


Fig. 5.1: Original 3D models (a) bunny, (b) car, (c) elephant, (d) face, and (e) twisted

To assess the viability of proposed watermarking scheme, sequences of experiments are executed to analyze the robustness and imperceptibility of watermarking method. In our watermark insertion processes, the step size for parameter was used as $\Delta k=0.001$. The step size was experimentally selected considering the tradeoff between execution time of embedding process and imperceptibility of the watermarked model. Therefore, lower the

Δk used(close to 0) ,lower is the distortion produced and higher is the execution time and vice versa for large Δk i.e. close to 1.

Descriptions of them are demonstrated in Table 5.1:

Table 5.1 Characteristics of 3d mesh models

Models	***No. of vertices***	***No. of Faces***
Bunny	35947	69451
Elephant	24955	49918
Car	988	1763
Face	299	562
Twisted	800	1600

Optimum length of watermark for any model to avoid distortion and maintain robustness as well is experimentally found out to be in the range less than equal to approximately one by hundredth times of the number of vertices present in the model.

Table 5.2 shows the embedding watermark strength factor α used in our experiment. The proposed algorithm is compared with the algorithm of Cho et al. [1].

Table 5.2 Watermark strength applied

Models	***Strength Factor***
Bunny	0.11
Elephant	0.12
Car	0.28
Face	0.15
Twisted	0.07

5.1. Evaluation of Imperceptibility

The first sought-after requirement of a watermarking method is the imperceptibility of the watermark. Therefore, after embedding the watermark bits in the original models, we need to evaluate the imperceptibility of the watermarking method. The quality of the watermarked models is measured by Vertex Signal-to-Noise Ratio (VSNR) which finds the visual difference between the original and the watermarked models. VSNR can be calculated using:

$$SNR = \frac{\sum_{i=1}^{N}(X_i^2+Y_i^2+Z_i^2)}{\sum_{i=1}^{N}\left[(X_i-X_i')^2+(Y_i-Y_i')^2+(Z_i-Z_i')^2\right]} \quad (19)$$

Where N is the number of vertices in the 1-ring neighborhood of the centre vertex including the centre vertex, (X_i, Y_i, Z_i) and (X_i',Y_i', Z_i') are the Cartesian coordinates of the vertex Vi before and after the watermark inserting, respectively.

$$\text{V-SNR}=20\log_{10}(\text{SNR}) \quad (20)$$

The other measure is the technique proposed by Cignoni et al. in Ref. [32] referred to as Root Mean square error (RMSE) which based on a correspondence between each pair of vertices of the models to compare; therefore it's restricted to the comparison between two meshes sharing identical connectivity. The root mean square error is evaluated as:

$$RMSE = \sqrt{\frac{\sum_{J=1}^{N}\left\|V-V_j^*\right\|^2}{N}} \quad (21)$$

The values of VSNR and RMSE for the examined models after embedding the watermark in the vertices that come out from utilizing proposed technique are shown in Table 5.3. It shows that the proposed method gives the greater value of VSNR and the lowest value of RMSE compared to proposed method I.

In the simulations, we inserted watermark into a mesh model considering the tradeoff between the robustness and the transparency of watermark. Then, vertex norms were divided into bins and one bit of watermark was hidden in each bin.

Table 5.3 Performance of watermarked meshes under no attack

Models	***Cho Approach[1]***		***Proposed Approach***	
	RMSE (x 10^{-3})	***VSNR***	***RMSE* (x 10^{-3})**	***VSNR***
Car	0.633	137.71	0.134	181.90
Face	0.269	173.14	0.077	238.27
Twisted	0.172	162.38	0.086	217.41
Elephant	0.491	142.70	0.067	217.16
Bunny	2.678	108.33	1.950	167.32

The table shows that the statistical approach used in the Cho approach cannot insert a watermark bit into every bin in the case of extremely small size models like face. This is primarily caused by the very fact that some of the bins are empty or don't contain enough vertices. For this reason, the Cho proposed methods aren't recommended to be applied to such small size models (approximately having fewer than 2000 vertices). And additionally the hidden watermark can be extracted absolutely from all stego models apart from the smallest size model. This suggests the Cho proposed methods guarantee to hide a watermark bit into each bin for models with a adequate range of vertices. Some artifacts appear in smooth regions like in the rump of the bunny. In particular, the artifacts are conspicuous in the flat regions, even when small strength factor is applied. This is primarily because of the very fact that each vertex is modified while not considering local curvature of the models. It's also caused by discontinuities in the boundaries of neighbor bins once the distribution is changed. As results, the Cho methods aren't applicable to CAD models with a flat region.

From the viewpoint of watermark transparency, proposed method maintains better visual quality than Cho proposed method. Proposed method has also taken care of the local curvature of the models by calculating feature vector of 1-ring neighborhood and thereby not selecting too flat regions for watermark.

The proposed algorithm can completely resist rotation, translation, uniform scaling and vertex reordering attacks. It is noticed that our approach has better correlation value than those of Cho in a series of attacks. In addition, because of the increased watermark

strength, the robustness for mild noise and smoothing attacks is quite strong so that the correlation value between the extracting watermark and the original watermark is 1. The greater the attack intensity is, the larger the distortion of 3D models is.

5.2. Attack-Centric Investigation

The attacks make up a significant factor when designing 3D mesh watermarking algorithms. In this section, we carefully discuss three types of attacks and introduce the presented solutions in the literature-

Geometric Attacks

These types of attacks only modify the geometric part of the watermarked mesh. No matter what is the nature of the geometric change, the attack is reflected by a change in vertices position.

5.2.1. Similarity Transformations.

Similarity transformation is considered to be a common operation instead of an attack, against which even a fragile watermark ought to be ready to withstand. It includes translation, rotation, uniform scaling, and the combination of the above three operations. Generally speaking, there are three different methods to make a watermark that's resistant to this attack. The primary solution is to use some primitives that are invariant to similarity transformations. Practically, these primitives are all some relative measures between several absolute and individual ones, and they embody the similarity between different meshes. The similarity transformation, like its name, will invariably keep these relative measures unchanged. Fortuitously, not only the watermarking primitives are kept unchanged, but also most synchronization schemes are insensible to this kind of attack. The solution is to watermark in an invariant space. One such space may be obtained by doing the subsequent steps.

1. Translate the origin of the coordinate system to the mesh gravity centre.

2. Calculate the principal axes of the mesh and rotate the object so they coincide with axes of the frame of reference.

3. Do a uniform scaling so the entire mesh is bounded in a unit sphere/cube. Then the watermark is inserted in this new space.

However the causality problem arises as a result of the variables utilized in precedent steps, like the gravity centre and principle axes orientations are most likely modified after watermark insertion. Thus there'll probably exist some extent of errors once reconstructing this space at the extraction.

To evaluate the robustness of our methods against distortion-less attacks, vertex rearrangement and similarity transforms were carried out. Vertex rearrangement attack was performed iteratively a hundred times, also changing the seed of random range generator for each iteration. Similarity transforms were applied with several combos of rotation, uniform scaling, and translation factors. It’s not necessary to tabulate watermark detection performance because both methods absolutely extracted the hidden watermark info. The implementation is totally invariant to uniform scaling and affine attacks. The modification in these parameters doesn't have an effect on the relative orientation of the normals at the vertices and therefore the local smoothness measure for every vertex remains unchanged. Moreover because the rotation and translation transformations solely change the placement of the model not the actual contents, the watermarked model is safe thus our algorithm provides high correlation between original and extracted watermarks. As intended, the proposed watermarking methods are absolutely robust against distortion-less attacks.

5.2.2. Signal Processing Attacks

A mesh can be considered as a signal in a three dimensional space. There are counterparts of the traditional one-dimensional signal processing techniques for 3D meshes, such as random additional noise, smoothing, enhancement, and lossless compression (usually realized by quantization). Although these operations may be very harmful to inserted watermarks, they are really common manipulations in animation and

special effects applications. Random noise, smoothing, and enhancement can be modeled in the spectral domain by a modification of the high-frequency part.

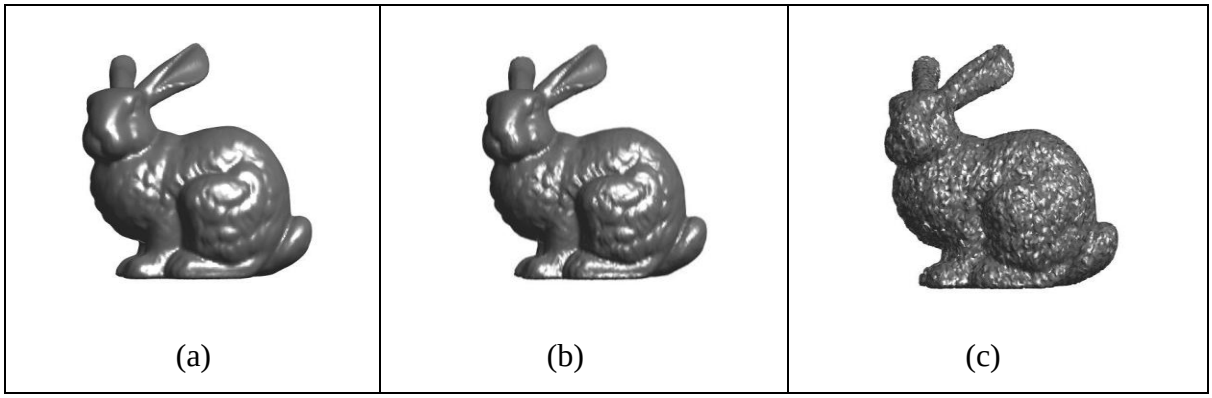


Fig. 5.2 (a)Original Bunny model watermarked by proposed method and attacked by (b) adding binary noise with error ratio of 0.5%, and(c) smoothing with iteration of 50 and relaxation of 0.1

Robustness of the 3D mesh model is evaluated by determining correlation between the original watermark and the detected watermark using:

$$Corr = \frac{\sum_{n=0}^{N-1}(\widetilde{w_n}-\bar{\widetilde{w}})(w_n-\bar{w})}{\sqrt{\sum_{n=0}^{N-1}(\widetilde{w_n}-\bar{\widetilde{w}})^2 \sum_{n=0}^{N-1}(w_n-\bar{w})^2}} \quad (22)$$

Where w and $\widetilde{w}$ are watermarking bits in the cover mesh model and the stego model of N number of vertices.

To evaluate the robustness of the watermark, various attacks were performed on the watermarked mesh model. Each attack was applied with varying attack strengths. Distortion attacks including additive binary random noise, smoothing, and simplification were carried out. As for instance, watermarked bunny models deformed by various distortion attacks are shown in Figure. 5.2. Noise was added to each vertex of watermarked model with three different error rates: 0.1%, 0.3%, and 0.5%. Here, the error rate represents the noise amplitude as a fraction of the maximum vertex norm of the object. We perform each noise attack five times using different random seeds and report the mean as shown in Table 5.4.

Table 5.4: Robustness evaluation against additive noise attacks

Models	***Error Rate***	***Correlation***	
		Cho Approach[1]	***Proposed Approach***
Car	0.10%	0.447	0.877
	0.30%	0.308	0.635
	0.50%	0.208	0.583
Face	0.10%	0.745	1
	0.30%	0.316	0.747
	0.50%	0.2	0.616
Twisted	0.10%	0.577	0.945
	0.30%	0.408	0.747
	0.50%	0.356	0.656
Elephant	0.10%	0.897	1
	0.30%	0.336	0.739
	0.50%	0.222	0.529
Bunny	0.10%	0.869	1
	0.30%	0.613	1
	0.50%	0.469	0.743

It is observed that our approach has better correlation value than those of Cho in a series of attacks. In addition, because of our improvised and increased watermark strength factor, the robustness against mild noise and smoothing attacks is quite strong so that the correlation value between the extracting watermark and the original watermark is better than Cho's method. Along with the increase of the attack strength, the accuracy of the extracted watermark has the different degree of reduction under the condition of common attacks.

Our method is reasonably resistant to the noise attacks under an error rate of 0.3, but good watermark detection can't be expected for higher error rates. This is due to the fact that the additive noise essentially modifies the distribution of vertex norms in the divided bins. In addition to this, more vertex norms exceed the range of each bin as the noise error rate increases. Similar tendency was observed in smoothing attacks. Due to these reasons, the robustness can't be improved beyond a certain level even when the strength factor increases. The robustness can also be improved by increasing the size of the bin, but the transparency of watermark and the number of bits to be inserted (i.e. capacity of embedding watermark) should be kept in mind.

Table 5.5 shows the performance of the watermarking schemes after smoothing attacks. Three different pairs of iteration and relaxation were applied. An example of this attack can also be seen in Figure 5.2 where the effect can be seen in the watermarked bunny model. The robustness depends on the smoothness of the original meshes.

Laplacian smoothing is applied to the watermarked model which smoothes the sharp edges in the model by applying a low pass gradient filter to the vertices.

5.2.3. Local Deformation Attacks.

A local deformation is sometimes imperceptible, but if we do not possess the original mesh for comparison, it can acutely disturb the watermark, especially the synchronization process. One innate solution is to divide the mesh into several patches and repeat the watermark embedding process in each patch. This division can be based on curvature or semantic analysis. As mentioned before, division into patches can also decrease the insertion time for some spectral techniques.

5.2.4. Connectivity Attacks

This category of attacks includes cropping, re-meshing, subdivision and simplification. In general, they're quite tough to handle. Cropping is a special attack and a few researchers opt to regard it as a geometrical attack because its consequence is kind of the same as the one caused by local deformation. Watermark repetition in several patches looks the most economical way to resist cropping. As far as the other attacks the algorithms that take the

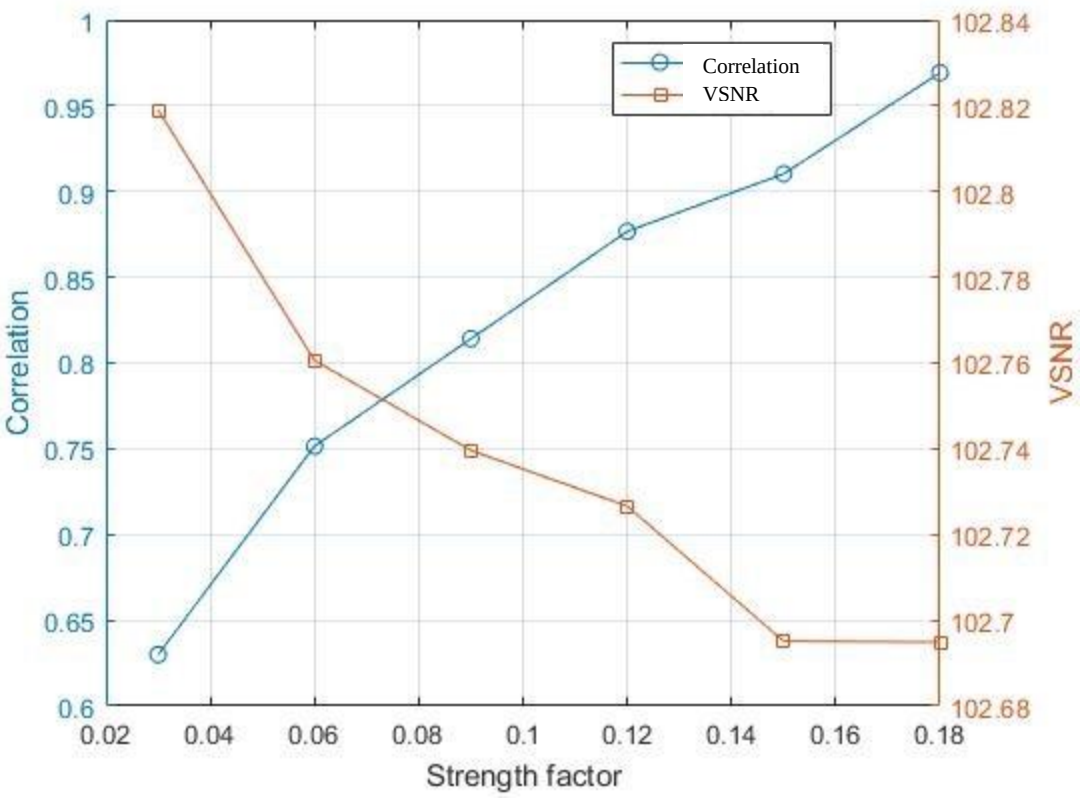
average normal direction of a group of facets or the distances of a group of vertices to the mesh centre as primitives, appear less smart.

Table 5.5: Robustness evaluation under smoothing attack

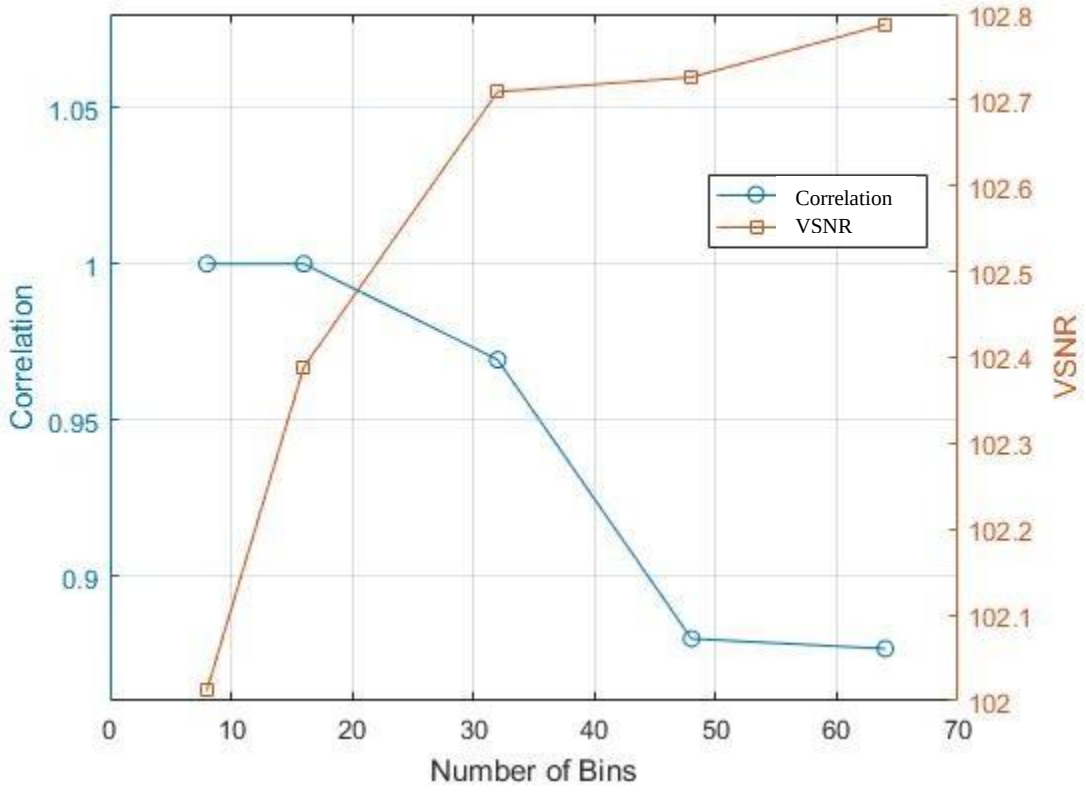
Models	***(# of iterations, relaxation)***	***Correlation***	
		Cho Approach[1]	***Proposed Approach***
Car	(10, 0.1)	0.997	1
	(30, 0.1)	0.737	1
	(50, 0.1)	0.672	1
Face	(10, 0.1)	0.891	1
	(30, 0.1)	0.632	0.867
	(50, 0.1)	0.216	0.516
Twisted	(10, 0.1)	0.317	0.816
	(30, 0.1)	0.267	0.767
	(50, 0.1)	0.067	0.352
Elephant	(10, 0.1)	1	1
	(30, 0.1)	0.691	0.910
	(50, 0.1)	0.554	0.882
Bunny	(10, 0.1)	1	0.969
	(30, 0.1)	0.654	0.819
	(50, 0.1)	0.528	0.739

Their primitives are approximately conserved after connectivity modification. Other spatial techniques are less robust by reasons of both the geometric modification of the primitives and the de-synchronization problem. The basis function construction and the frequency coefficients calculation in direct spectral analysis are either dependent to vertices order or to mesh connectivity. The existing multi-resolution analysis tools often have connectivity restrictions, and also the re-meshing step isn't robust enough to connectivity modification. So, to achieve a adequate robustness for these methods, the

authors usually suggest doing a pre-processing step of connectivity restoration before extraction. This restoration procedure can be thought of as a re-sampling of the extraction input mesh (objective mesh) so as to acquire the same connectivity configuration as the cover mesh or the non-attacked stego-mesh (reference mesh). The task is to seek out, for every vertex within the reference mesh, a corresponding point on the surface of the targeting to reduce a objective mesh. This correspondence can be established by the nearest neighbor criterion, ray intersection , or iterations particular cost function. 2 other possibilities to handle connectivity attacks are to find a robust transformation or parameterization domain that's independent to connectivity, and to watermark some robust mesh shape descriptors.



(a)



(b)

Fig. 5.3 Relationship (a) between the strength factor and the correlation and (b) between the no. of bins and the correlation. A noise attack with 0.3% noise amplitude is used as an example

Considering the tradeoff between the robustness and the transparency of watermark, we defined strength factor α as the maximum deviation of mean across all bins from 0.5 for a particular model. High value of α on one hand increases robustness but deteriorates the imperceptibility requirement of watermarked model, whereas low value of α retains imperceptibility but decreases robustness towards some common attacks as shown in Fig. 5.3(a). Number of bins selected for insertion too has an impact on the robustness as well as imperceptibility of the watermarking process as shown in Fig. 5.3(b). If we increase the number of bins, correlation between inserted watermark and the retrieved one gets poorer because binning process becomes ineffective when number of bins increases. However increase in number of bins has a slight improvement in VSNR making watermark less perceptible.

CHAPTER 6 CONCLUSION

Three-dimensional mesh watermarking seems as a noteworthy and promising research area. We can imagine several potential practical applications of 3D model/graphics watermarking. as an example, an automobile constructor may insert watermarks within the automotive elements it has designed to guard its intellectual properties; a doctor may hide a patient's personal data within the 3D mesh model obtained after a scan, while not impacting his diagnosis, to avoid mismatching patient's personal data and his scan result; a mesh data receiver could certify the integrity and originality of the mesh model he/she has bought or obtained; even the texture of a mesh model, or the motion parameter of a mesh sequence can be inserted within the mesh description file via watermarking, similar to concealing the audio signal of a video inside the visual a part of the video stream. However, because of several difficulties expressed in review like the irregularity of the mesh description and also the quality of the potential attacks, the analysis work on 3D mesh watermarking continues to be in its infancy, even after 10 years of studies by several contributors. For fragile techniques of arbitrary meshes, constructing an algorithm capable of accurately locating the endured attacks and capable of surviving similarity transformations and vertex rearrangement could be a tough task. For robust techniques, the causality problem, the de-synchronization problem and also the attacks (especially the connectivity attacks) are not really easy to handle. We've provided some working directions to making robust and blind algorithms. Nearly all of them rely on a supposed efficient analysis or description tool of 3D meshes. They include global or local mesh shape descriptors, robust mesh transformations, and re-meshing algorithms insensitive to varied attacks. Thus, in our opinion, the most necessary, also the toughest part of a 3D mesh watermarking system is that the choice of an appropriate feature space, within which the watermark signal is inserted. Thus to attain this target, the watermarkers most likely ought to work closely with computer graphics and geometry processing experts.

In this thesis, we tend to propose a blind and robust 3D watermarking algorithm based on vertex smoothness measure and piecewise mapping function to enhance the robustness

whereas guaranteeing a good performance in the transparency. On the one hand, the algorithm reasonably divides chosen vertices into corresponding bins to reinforce the robustness of the watermark in the marginal bins by feature vector calculation and their binning into eight bins. On the other hand, the algorithm adjusts the mean of norm of chosen vertices to insert the watermark by the piecewise mapping function, that ensures the low distortion of the model. The experiment results demonstrate that the proposed method is more robust against common attacks like similarity transformation and random noise attacks compared with the Cho's methodology. However, there are some drawbacks. Our proposed methodology is extremely prone to cropping and clipping attacks that cause severe alteration to the center of gravity of the model and through simulations we tend to realized the performance of our proposed methodology in such attacks to be less than or equal to Cho methodology. Nonetheless, the simulation results demonstrate a possible, oblivious statistical watermarking method based on vertex smoothness measure for 3-D polygonal mesh model.

In future work we propose to use an adaptive approach to divide the bins based on feature vector considering number of vertices and their 1-ring neighbourhood .

CHAPTER 7 REFERENCES

[1] J. Cho, R. Prost and H. Jung, "An Oblivious Watermarking for 3-D Polygonal Meshes Using Distribution of Vertex Norms", *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 142-155, 2007.

[2] M. M. Soliman, A. E. Hassanien, and H. M. Onsi, “Robust watermarking approach for 3D triangular mesh using self organization map,” *2013 8th International Conference on Computer Engineering & Systems (ICCES)*, 2013.

[3] O. El Zein, L. El Bakrawy and N. Ghali, "A robust 3D mesh watermarking algorithm utilizing fuzzy C-Means clustering", *Future Computing and Informatics Journal*, vol. 2, no. 2, pp. 148-156, 2017.

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

[5] O. Benedens, "Geometry-based watermarking of 3D models", *IEEE Computer Graphics and Applications*, vol. 19, no. 1, pp. 46-55, 1999.

[6] S. H. Lee, T. S. Kim, B. J. Kim, S. G. Kwon, K. R. Kwon, and K. I. Lee, “3D polygonal meshes watermarking using normal vector distributions,” in *IEEE Int. Conf. Multimedia Expo*, Jul. 6–9, 2003, vol. 3, pp. 105–108

[7] Z. Yu, H. S. Ip, and L. F. Kwok, “A robust watermarking scheme for 3D triangular mesh models,” *Pattern Recognit*., vol. 36, no. 11, pp. 2603–2614, 2003

[8] S. Kanai, H. Date, and T. Kishinami, “Digital watermarking for 3D polygons using multiresolution wavelet decomposition,” *in Proc. 6th IFIP*, Tokyo, Japan, Dec. 1998, pp. 296–307, WG 5.2, GEO-6

[9] K. Yin, Z. Pan, J. Shi, and D. Zhang, “Robust mesh watermarking based on multiresolution processing,” *Comput. Graph*., vol. 25, pp. 409–420, 2001

[10] R. Ohbuchi, S. Takahashi, T. Miyazawa, and A. Mukaiyama, “Watermarking 3D polygonal meshes in the mesh spectral domain,” *in Proc. Graph. Interface*, Ottawa, ON, Canada, Jun. 2001, pp. 9–17

[11] D. Cotting, T. Weyrich, M. Pauly, and M. Gross, “Robust watermarking of point-sampled geometry,” *in IEEE Int. Conf. Shape Modeling Int. 2004*, 2004, pp. 233–242.

[12] R. Ohbuchi, A. Mukaiyama, and S. Takahashi, “Watermarking a 3D shape model defined as a point set,” *in IEEE Int. Conf. Cyber Worlds 2004*, Tokyo, Japan, Nov. 2004, pp. 392–399.

[13] A. Kejariwal, “Watermarking,” IEEE Potentials, pp. 37–40, Oct./Nov. 2003. [21] P. H. W. Wong, O. C. Au, and Y. M. Yeung, “A novel blind multiple watermarking technique for images,” *IEEE Trans. Circuits Syst. Video Technol*., vol. 13, pp. 813–830, Aug. 2003.

[14] P. H. W. Wong, O. C. Au, and Y. M. Yeung, “*A novel blind multiple watermarking technique for images*,” IEEE Trans. Circuits Syst. Video Technol., vol. 13, pp. 813–830, Aug. 2003.

[15] S. Craver, N. Memon, B. L. Yeo, and M. Yeung, “Can invisible watermarks resolve rightful ownerships*?,” in Proc. IS&T/SPIE Conf. Storage Retrieval Image Video Database V, San Jose*, CA, Feb. 13–14, 1997, vol. 3022, pp. 310–321.

[16] W. G. Kim, J. C. Lee, and W. D. Lee, “An image watermarking scheme with hidden signatures*,” in Proc. IEEE Int. Conf. Image Process.*, Kobe, Japan, Oct. 24–28, 1999, vol. 2, pp. 206–210.

[17] E. Praun, H. Hoppe, and A. Finkelstein, “Robust mesh watermarking,” *in Proc. SIGGRAPH99*, Los Angeles, CA, Aug. 1999, pp. 49–56

[18] J. Jian-qiu, D. Min-ya, B. Hu-jun, and P. Qun-sheng, “Watermarking on 3D mesh based on spherical wavelet transform,” *in JZUS, 2004*, vol. 5, no. 3, pp. 251–258

[19] J. W. Cho, M. S. Kim, R. Prost, H. Y. Chung, and H. Y. Jung, “Robust watermarking on polygonal meshes using distribution of vertex norms,” i*n Digital Watermarking (LNCS3304)*, Mar. 2005, pp. 283–293

[20] Y. Zhang, C. Wang, X. Wang and M. Wang, "Feature-Based Image Watermarking Algorithm Using SVD and APBT for Copyright Protection", *Future Internet*, vol. 9, no. 2, p. 13, 2017.

[21] X. Feng, Y. Liu and L. Fang, "Digital Watermark of 3D CAD Product Model", *International Journal of Security and Its Applications*, vol. 9, no. 9, pp. 305-320, 2015.

[22] M. Soliman, A. Hassanien and H. Onsi, "A Blind 3D Watermarking Approach for 3D Mesh Using Clustering Based Methods", *International Journal of Computer Vision and Image Processing*, vol. 3, no. 2, pp. 43-53, 2013.

[23] P. Flynn and A. Jain: “On Reliable Curvature Estimation,” Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition, pp. 110-116, 1989.

[24] O. Benedens: “Robust Watermarking and Affine Registration of 3D Meshes,” Proc. of 5th International Workshop on Information Hiding, Netherlands, October 7-9, pp. 177-195, 2002

[25] Q. Ai, Q. Liu, Z. Zhou, L. Yang and S. Xie, "A new digital watermarking scheme for 3D triangular mesh models", *Signal Processing*, vol. 89, no. 11, pp. 2159-2170, 2009.

[26] J. Hou, D. Kim and H. Lee, " A non-blind robust watermarking approach for 3d mesh models ", *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2712-2725, 2017.

[27] I. Prathap and R. Anitha, "Robust watermarking approach for 3D Triangular Mesh using Self Organisation Map", *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 51-58, 2014.

[28] A. Bors and Ming Luo, "Optimized 3D Watermarking for Minimal Surface Distortion", *IEEE Transactions on Image Processing*, vol. 22, no. 5, pp. 1822-1835, 2013.

[29] K. Kim, M. Barni and H. Tan, "A watermarking scheme for 3D mesh models using Haar Discrete Wavelet Transform", *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 4, pp. 721-733, 2010.

[30] Z. Yu, H. Ip and L. Kwok, "A 3D mesh watermarking based on improved vertex grouping and piecewise mapping function", *Pattern Recognition*, vol. 36, no. 11, pp. 2603-2614, 2003.

[31] S. Zafeiriou, A. Tefas and I. Pitas, "Blind Robust Watermarking Schemes for Copyright Protection of 3D Mesh Objects", *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, no. 5, pp. 596-607, 2005.

[32] E. Abdallah, A. Ben Hamza and P. Bhattacharya, "Watermarking 3D models using spectral mesh compression", *Signal, Image and Video Processing*, vol. 3, no. 4, pp. 375-389, 2008.

[33] Y. Zhu, "Analysis and Simulation on SVD-Based 3D Mesh Digital Watermark Algorithm", *Advanced Materials Research*, vol. 846-847, pp. 1052-1055, 2013.

[34]]K. Wang, G. Lavoué, F. Denis and A. Baskurt, "Robust and blind mesh watermarking based on volume moments", *Computers & Graphics*, vol. 35, no. 1, pp. 1-19, 2011.

[35] Cignoni, P., Rocchini, C. and Scopigno, R. (1998). Metro: Measuring Error on Simplified Surfaces. *Computer Graphics Forum*, 17(2), pp.167-174.

[36] A. Molaei, H. Ebrahimnezhad and M. Sedaaghi, "Robust and Blind 3D Mesh Watermarking in Spatial Domain Based on Faces Categorization and Sorting", *3D Research*, vol. 7, no. 2, 2016.

[37] M. Dorairangaswamy, "Robust Blind Image Watermarking Scheme in Spatial Domain for Copyright Protection", *International Journal of Engineering and Technology*, vol. 1, no. 3, pp. 249-255, 2009.

[38] S. Valette, A. Gouaillard and R. Prost, "Compression of 3D triangular meshes with progressive precision", *Computers & Graphics*, vol. 28, no. 1, pp. 35-42, 2004.

[39] S. Yang and Z. Yao, "A Data Hiding Scheme based on Local Coordinate System for 3D Triangle Mesh Models", *Journal of Software*, vol. 5, no. 4, 2010.

[40] S. Gayathri and D. Venkatesan, "Particle Swarm Optimization and Discrete Wavelet Transform based Robust Image Watermarking", *Indian Journal of Science and Technology*, vol. 9, no. 48, 2016.

[41] L. Li, D. Zhang, Z. Pan, J. Shi, K. Zhou and K. Ye, "Watermarking 3D mesh by spherical parameterization", *Computers & Graphics*, vol. 28, no. 6, pp. 981-989, 2004.

[42] Y. Zhan, Y. Li, X. Wang and Y. Qian, "A blind watermarking algorithm for 3D mesh models based on vertex curvature", *Journal of Zhejiang University SCIENCE C*, vol. 15, no. 5, pp. 351-362, 2014.