International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org
Volume 4, Issue 6, November - December 2015 ISSN 2278-6856

Segment based watermarking in 3D triangular mesh

Lokendra Kumar Sharma[1] and Dr. D. B. Ojha[2]
[1]Research Scholar, Department of CSE, Mewar University, Chittorgarh, Rajasthan, India
[2]Director Research, Mewar University, Chittorgarh, Rajasthan, India
[1]lokendra_sharma@hotmail.com [2]ojhabrat@gmail.com

***Abstract*—3D mesh objects are one of the category of 3D objects. In this work, vertices are selected for watermark insertion based on code generated from cryptographic function SHA-256. The same watermark information is redundantly inserted into the 3D mesh object. Authentication can be done from any of the segment in case of any attack. The proposed algorithm is based on geometrical property of the vertices where vertices are shifted from their original position as per output of cryptographic function SHA-256. The change in position and selected vertices are the watermark information. The Cartesian coordinate of vertices are converted into spherical coordinate. The vertices are shifted from its original position by displacing vertex normal distance (ρ) component of (ρ,θ,ϕ) spherical coordinate of vertices.**

Keywords*—3D triangular Mesh Objects, Hausdorff distance, RMS, secret key

I. Introduction

In the proposed work, we are focusing on security of 3D objects. Extensive market growth of 3D object in last decades focuses attention on security issues of these objects. 3D object are widely used in architecture design, machine design, cultural heritage[1] and entertainment. 3D objects specialized data which can be viewed any any angle of object i.e we can see the object from front, back, side, top, lower or any other view. These 3D objects are basically categorized into two classes: Volume based or surface based objects. In the proposed work, we are considering 3D triangular mesh objects, special class of surface base objects.

In watermarking, a piece of digital information called watermark, is embedded into cover data in a persistent way by modifying some characteristics of digital host data. In the case of copy-right protection, the embedded watermark gives the information about the owner of the digital host data and can be used to prove the ownership in legal dispute. The applications other than the copyright protection are copy control, authentication, annotation, tracking, content distribution, fingerprinting, error recovery in multimedia transmission, labeling for data retrieval and linking real objects to the digital world etc.

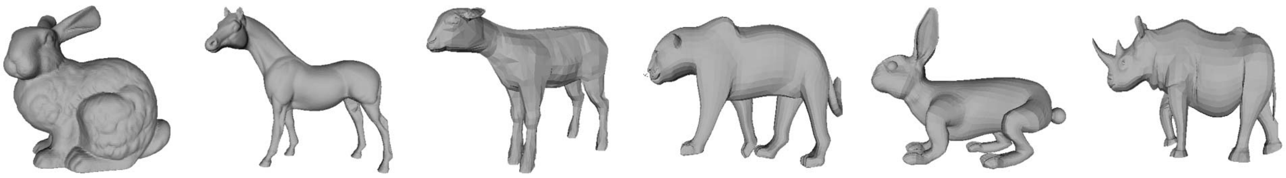
Watermarking is an art of hiding secondary data on the primary data by maintaining perceived quality of primary data[2]. The watermarking scheme is designed to check the copy-right ownership (robust watermarking) or to verify the authentication (fragile watermarking)[3]. Watermarking techniques are being used for copy right protection and the integrity of digital contents. 3-D modeling is a process of developing a mathematical representation of any 3-D surface of object(either inanimate or living) via specialized software like CAD/CAM. The product is called a 3-D model or 3-D object. These 3D triangular mesh objects are categorized by these surface property while objects which are categorized by their volume are called volume based objects[4]. The review of current 3-D watermarking techniques[5] motivates to develop a watermarking technique in spatial domain which should be robust, non-blind, imperceptible and secure. Different watermarking algorithms have been summarized [6] with different pros and cons in spatial and spectral domain. Spatial domain watermarking algorithms produce output of better quality[7] than watermarking algorithms of spectral domain. The non-blind nature of an algorithm makes it robust[5] [8] against different types of attacks. Usually, non-blind watermarking algorithms are used for authentication purpose.

In 3D triangular mesh objects vertices and faces are geometrical and topological attributes respectively. We consider vertices for watermark embedding as watermark insertion using geometrical attributes are much robust as compare to the topological attributes[9].

II. WATERMARKING ALGORITHM

3D polygon mesh is a group of geometrical and topological data. Vertices are geometrical data while edge, face, vertices normal are topological data. 3D polygon mesh can be represented by G $\equiv$ (V,F) where V is a finite set of vertices and F is a finite set of faces[10]. 3D triangular mesh is specialized from of 3D polygon mesh where each vertices are connected to form the triangular face. The triangular face is the smallest unit of the face. Change in topological data does not modify the shape of the 3-D mesh while it is not true for geometrical data[11].

In the proposed watermarking scheme watermark is embedded into 3D triangular mesh by modifying the vertex normal distances of selected vertices. The perceivable visual quality of 3D triangular mesh is preserved by inserting watermark with some gaps instead of selecting all vertices. The vertices are selected for watermark embedding as per the secret code obtained by applying SHA-256 function on some secret key. The SHA-256 generate 256 bits code which is repeatedly



(a) Bunny (b) Horse (c) Lamb (d) Leopard (e) Rabbit (f) Rhino

Fig. 1: Original 3-D mesh objects

Algorithm 1 Segment Based Watermarking in 3D Triangular mesh

INPUT: 3D triangular mesh object having n vertices and Secret Key
OUTPUT: Watermarked 3D object

```
Let $m = n/256$
Group the n vertices of 3D triangular mesh object into m groups
Apply SHA256 function on a secret key to obtain a secret code of 256 bits ($p_1$,$p_2$ ...... $p_{256}$)
Calculate the centre of mass coordinates ($x_{cm}, y_{cm}$ and $z_{cm}$) of 3D triangular mesh object
Convert cartesian coordinate of the vertices ($x_i$, $y_i$, $z_i$) of the Original 3D triangular mesh ($O$) into spherical coordinate
Select the vertices for watermark embedding in each of the m segment, according to secure code ($p_1$,$p_2$ ...... $p_{256}$)
for selected verteices from each group do
    Modify vertex using $\rho_i' = \rho_i + W$
    Re-convert the spherical coordinate into cartesian coordinate to obtain watermarked object
end for
```

Algorithm 2 *Watermark authentication in 3D triangular mesh*

INPUT: Original 3D triangular mesh O, watermarked 3D object(O^w), original secret key
OUTPUT: Authentication result

```
for each vertex of 3D triangular mesh ($O$) and watermarked 3D object ($O^w$) do
    Convert cartesian coordinate of the vertices ($x_i$, $y_i$, $z_i$) into spherical coordinate
    Calculate the vertex normal distances of watermarked object w.r.t center of mass of original object
    Assign non zero value of the difference as 1 to obtain sequence of bits (1/0)
end for
Compare the sequence of bits (1/0) with secret code obtained from one secret key and SHA256
if sequence of bits match with the hash code then
    the object is authenticated
else
    otherwise report tempering
end if
```

requires secret Key and watermark strength during watermark authentication.

III. Result Analysis

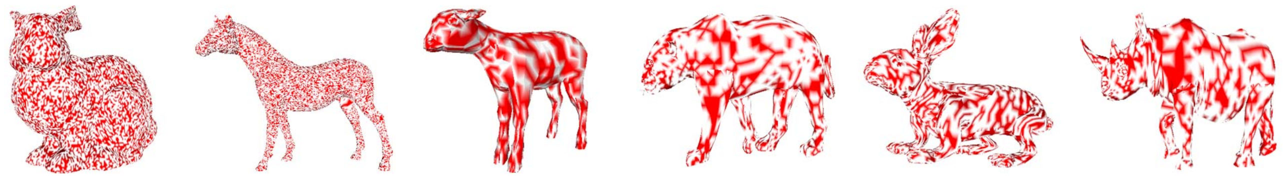
The watermarking algorithm is performed on different 3D mesh objects Bunny ($v = 34834, f = 69451$), Horse($v = 48484, f = 96964$), Lamb($v = 2995, f = 3960$), Leopard($v = 7302, f = 11508$), Rabbit($v = 14411, f = 14976$), Rhino ($v = 6459, f = 9374$) as shown in Figure-1. These object are taken from www.archive3d.net for experimental purpose. We have selected these 3D mesh objects due to variation in parameters (number of vertices, structure, surface roughness etc) of 3D mesh objects.

The distortion is measured between original mesh and watermark mesh. Hausdorff distance, Root Mean Square Error (RMS), surface roughness are basic parameters for measuring the distortion. Hausdorff distance, Root Mean Square Error (RMS) are measured by Metro Tool[13]. In the watermark mesh object, watermark information can be completely retrieved. Correlation factor 100% signifies that 100% of watermark vertices are identified correctly. Similarly, correlation factor is measured against different attacked watermark mesh objects. Robustness of an algorithm is directly proportional to the correlation factor. We have shifted the selected vertices from their original position by some fixed amount (W) and the amount of shifting of vertices must be less than the limiting value. The vertices of 3D mesh objects are shifted to 10%, 20% and 25% from their original value. The distortion increases by increasing the weight.

Bunny 3D mesh object has rough surface, Horse has smooth surface, Lamb has bumpy surfaces, Rabbit and Rhino has sharp pointed edges. These selected 3D mesh objects also varies in number of vertices and faces. Hausdorff distance and RMS are the parameters for evaluating the distortion introduced due to watermark insertion. The correlation factor is also determine on watermark model without any attack.

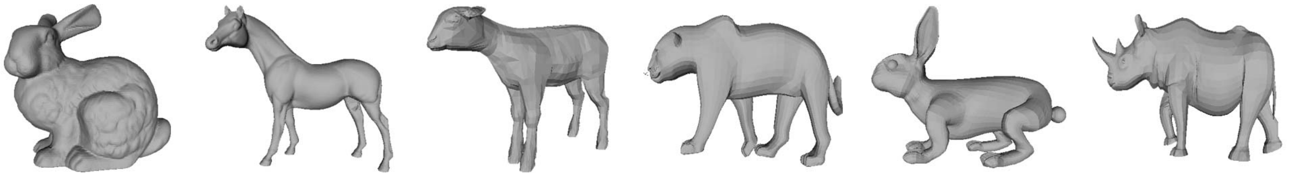
A. *Hausdorff Distance*

Hausdorff distance between two sets of points is defined as the maximum distance of a set to the nearest point in the other



(a) Bunny (b) Horse (c) Lamb (d) Leopard (e) Rabbit (f) Rhino

Fig. 2: Selected vertices of 3-D mesh objects indicated in red color



(a) Bunny (b) Horse (c) Lamb (d) Leopard (e) Rabbit (f) Rhino

Fig. 3: Watermarked 3-D mesh objects

set[14]. Hausdorff distance estimates the extent to which each point of a object set lies near some point of another object set and vice versa. This distance is used to estimate the degree of resemblance between two objects that are superimposed on one another. Informally, two sets of points are close if each point of either set is close to some point of the other set. The objective is to minimize the hausdorff distance to reduce the degree of mismatch between cover object(O) and watermarked object(O^w)[14] [15].
Let e(p,O) represent the distance of a point p in 3-D space from the 3-D object O as[6]:

$$e(p,O) = \min_{v_i^O \in O} \{d(p, v_i^O)\} \tag{9}$$

where $d(p, v_i^O)$ is the Euclidian distance between v_i^O, i^{th} vertex of object O and p. Hausdorff distance between two 3-D objects O and O^w is:

$$H_a(O,O^w) = \max_{v_i^O \in O} \{e(v_i^O, O^w\} \tag{10}$$

This distance is not symmetrical i.e $H_a(O,O^w) \neq H_b(O,O^w)$. $H_a(O,O^w)$ and $H_b(O,O^w)$ are referred as forward and backward distance respectively[6].
The symmetrical Hausdorff distance can be defined as:

$$H_d(O,O^w) = \max(H_a(O,O^w), H_a(O^w,O)) \tag{11}$$

The Symmetrical Hausdorff distance reports more accurate measurement of the error between two surfaces as computation of a ”one-sided” error can lead to significantly underestimated distance value[14].

B. Root Mean Square Error (RMS)

The Root Mean Square error is based on the correspondence between each pair of vertices of the objects to compare, thus it is limited to the comparison between two meshes sharing the same topology[6][14]. The root mean square error is evaluated as :

$$d_{rms}(O,O^w) = \sqrt{\sum_{i=1}^{n} ||v_i^O - v_i^{O^w}||^2} \tag{12}$$

where n is number of vertices of mesh and v_i^O is a vertex of O corresponding to the vertex $v_i^{O^w}$ of O^w. The attacks on 3D mesh objects are entirely different from the attacks on images. Image attacks include up sampling, down sampling, cropping, noise, compression, rotation, scaling, translation, filtering etc while 3D mesh object attacks includes subdivision, smoothing, simplification, rotation, scaling, translation, vertex reordering, face reordering, re-meshing etc. The robustness of any watermarking algorithm is measured in terms of watermark information exist even after applying different attacks. In 3D watermarking, we are applying watermarking based on geometrical properties rather than topological properties, therefore re-meshing, face reordering will not effect the watermark information. Subdivision, smoothing, simplification attacks are based on geometrical character of 3D mesh therefore robustness is measured against these attacks.

Moreover, 3D attacks can also be categorized based on distortion produced after applying different attacks. The attacks which introduces perceivable distortion are said be distortion attack while rest are called distortion less attacks. Subdivision, smoothing, simplification are distortion attacks while rotation, scaling, translation, vertex reordering, face reordering, re-meshing are distortion-less attacks. In robustness evaluation, correlation factor is measured which reports the watermark information extracted. The non-blind watermarking are expensive in terms of execution time but are robust against vertex reordering, transformation, rotation, uniform scaling and transformation.

C. Distortion Attack

In distortion attacks watermark information extracted is not reported at higher rate as these attacks disturb the geometry of the 3D mesh object by increasing or decreasing the number of vertices. Therefore, correlation factor also decrease significantly.

Mesh smoothing is a geometrical process that is generally done on the 3D meshes achieved through 3D scanning. Generally, 3D scanners produce ”noisy” surfaces due to the approximation errors introduced during the surface reconstruction process. Smoothing reduces the surface level by removing some vertices as per the values of their neighboring vertices. The performance of smoothing attack on the experimental models are shown in Table I. Smoothing attacks is applied in MeshLab open source which uses Taubin Smoothing filter. Correlation factor reports the watermark information authenticated after attacks.

Mesh simplification also known as decimation is related to the reduction of the number of vertices and faces of a 3D mesh object while maintaining its shape[11] [16]. It is considered to more destructive attack for 3D mesh objects. We have used Re-Mesh toolbox for applying simplification attack[16]. We have simplified the 3D mesh objects up to 25%, 50%, 75% and determine the correlation factor as shown in Table-II. The uniform distribution of watermark information from its original position make it robust.

Mesh subdivision enhances the object quality by increasing the density of the vertices on the surface iteratively. The performance of subdivision attack is shown in Table-III. The result reports considerably good degree of robustness in terms of correlation factor. The distortion is considered between original object and attacked watermark object as shown in Table-IV. It is observed that watermark information is retrieved 93%, 95%, 88%, 76%, 87%, and 72% from Bunny, Horse, Lamb, Leopard, Rabbit, and Rhino respectively after applying subdivision attack.

D. Distortion-less Attack

The non-blind nature of algorithm ensures the robustness against different distortion-less attacks. In non-blind watermarking both cover and watermarked objects are available for comparative analysis. The watermark insertion is invariant to Translation, Rotation and Uniform scaling (RST) as ratio of the distance between center of mass and vertex remains the same.

$$\rho_i' = \rho_i + W \tag{13}$$

After scaling by factor t vertex normal becomes $\rho_i'' = (\rho_i' * t)$ or

$$\rho_i'' = (\rho_i + W) * t \tag{14}$$

where ρ_i, ρ_i' and ρ_i'' represent the vertex normal distances of cover object, watermark object and attacked watermark object respectively.

The uniform scaling reflects the constant ratio between the normal distance of vertices of distorted and watermarked object, which is computable. Thus, the watermark information is not degraded by uniform scaling. Similarly, normal distance of the vertex from center of mass remains same preserving transformation and rotation attack. Using the property of constant ratio or distance of each vertex from center of mass of 3-D mesh, all the vertices are again rearrange by taking help from cover object and watermark object. In RST attack, we are able to retrieve 100% of watermark information. The non-blind nature of the proposed scheme makes the algorithm robust and secure against vertex reordering attack as both cover and watermarked objects are available for comparison analysis.

TABLE I: Robustness measure against smoothing attack

Model	$H_s(O,O^{w'})$	$d_{rms}(O,O^{w'})$	Correlation
Bunny	0.000823	0.60×10^{-4}	0.52
Horse	0.001128	0.43×10^{-4}	0.53
Lamb	0.030614	0.44×10^{-2}	0.57
Leopard	1.096693	0.11×10^{0}	0.38
Rabbit	1.602641	0.21×10^{0}	0.46
Rhino	0.131558	0.23×10^{-1}	0.42

TABLE II: Robustness measure against simplification attack

Model	Simplify	$H_s(O,O^{w'})$	$d_{rms}(O,O^{w'})$	Correlation
Bunny	25.00%	0.000849	0.31×10^{-4}	0.86
	50.00%	0.000947	0.33×10^{-4}	0.75
	75.00%	0.001407	0.43×10^{-4}	0.36
Horse	25.00%	0.000184	0.22×10^{-4}	0.90
	50.00%	0.000184	0.23×10^{-4}	0.63
	75.00%	0.000578	0.26×10^{-4}	0.32
Lamb	25.00%	0.056342	0.29×10^{-2}	0.95
	50.00%	0.056342	0.38×10^{-2}	0.83
	75.00%	0.063083	0.64×10^{-2}	0.41
Leopard	25.00%	1.810001	0.73×10^{-1}	0.71
	50.00%	1.810001	0.78×10^{-1}	0.52
	75.00%	1.726135	0.10×10^{0}	0.31
Rabbit	25.00%	2.014728	0.97×10^{-1}	0.91
	50.00%	3.329350	0.22×10^{0}	0.78
	75.00%	3.432703	0.33×10^{0}	0.33
Rhino	25.00%	0.218377	0.12×10^{-1}	0.74
	50.00%	0.258154	0.15×10^{-1}	0.51
	75.00%	0.357699	0.22×10^{-1}	0.28

IV. Comparative Assessment

The proposed algorithm is compared with Cho et al.[17] and Wang et al.[18] against robustness and distortion. Cho et al.[17] proposed two algorithms ChoMean and ChoVar based on mean and variance as described in [17]. Similarly, Wang