

A Fragile Watermarking Scheme for 3D Meshes

Hao-Tian Wu
Department of Computer Science
Hong Kong Baptist University
224 Waterloo Road, Kowloon Tong
Hong Kong, China
htwu@comp.hkbu.edu.hk

Yiu-Ming Cheung
Department of Computer Science
Hong Kong Baptist University
224 Waterloo Road, Kowloon Tong
Hong Kong, China
ymc@comp.hkbu.edu.hk

ABSTRACT

In this paper, we propose a new fragile watermarking scheme for 3D meshes. Firstly, the watermark information is adaptively embedded into the mesh geometry by slightly adjusting the vertex positions while the mesh topology is unchanged. The embedded watermark can be blindly retrieved with some priori knowledge and used to authenticate the integrity of the watermarked mesh. Compared with previous methods, our proposed one has the following advantages: (1) the embedded watermark is invariant to translation, rotation and uniformly scaling, but sensitive to other operations; (2) enhanced with a reference position, any processing made to the watermarked mesh can be detected; (3) the embedding strength is adjustable so that a trivial tampering with the watermarked mesh model may lead the embedded watermark to change. As a result, unauthorized modifications of the mesh model can be detected and classified.

Categories and Subject Descriptors

I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—*Geometric algorithms, languages, and systems*

General Terms

Algorithms

Keywords

fragile watermarking, mesh authentication, dither modulation, mesh centroid, tampering detection

1. INTRODUCTION

With the widespread use of 3D models from entertainment to engineering, science and medicine, such as CAD/CAM industry, art heritage and game industry, protecting 3D models from pirating or tampering, has received much attention in the community. As an effective complementary measure

to traditional encryption, digital watermarking for multimedia data (e.g. digital images, 3D models, video and audio streams) has been introduced and arduously studied to prove the ownership of digital works, verify their integrity, trace the content, etc.

Recently, considerable progress has been made in the area of watermarking 3D objects, such as 3D polygonal meshes and various 3D geometric CAD data. A variety of watermarking algorithms have been proposed in the spatial or frequency domain to embed the watermarks into 3D models, 3D mesh models in particular [3]. To prove the ownership of 3D models, most of the proposed algorithms attempt to embed the watermarks with robustness against those operations to which 3D models are routinely subjected, such as affine transformation, mesh simplification, etc.

Nevertheless, only a few fragile watermarking algorithms [4-8] have been proposed to authenticate the integrity of 3D models. Actually, the first fragile watermarking of 3D objects for verification purpose was addressed by Yeo and Yeung in [4], as a 3D version of the approach proposed for 2D image watermarking. A public key method to authenticate CSG models was proposed by Fornaro and Sanna in [6]. In [7], a fragile watermark immune to vertex reordering was embedded to detect malicious attacks. These algorithms rely on the vertex coordinate, a translation operation would easily change the embedded watermark. However, the integrity of the mesh model is not affected by translation, rotation and uniformly scaling operations. Therefore, it is advantageous to distinguish translation, rotation and uniformly scaling from other operations in authenticating 3D mesh models.

In this paper, we shall present a new fragile watermarking scheme to authenticate 3D mesh models. Our method is similar to one of the algorithms proposed in [8] called Vertex Flood Algorithm, which is suitable for 3D model authentication with respect to certain tolerances, e.g. truncation of mantissas of vertex coordinates. Basically, their algorithm modifies the vertices so that their distances to the center of a designated start triangle encode the watermark bits. In contrast, our algorithm modulates the distances from the mesh faces to the mesh centroid using dither modulation technique [2]. Compared to the Vertex Flood Algorithm, the embedded watermark using our method is more sensitive to modifications on the watermarked model, while less distortion of the mesh model has been introduced.

In our proposed method, the encoding process is adaptive to the original mesh to generate the watermarked mesh based on its global geometrical characteristics, while the

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

MM-SEC'05, August 1-2, 2005, New York, New York, USA
Copyright 2005 ACM 1-59593-032-9/05/0008 ...\$5.00.

mesh topology is not changed. Compared to the previous methods, our method makes the embedded watermark invariant to translation, rotation and uniformly scaling, but sensitive to other geometrical or topological operations. Enhanced with a reference position, any alteration of the watermarked mesh can be detected and classified. The rest of this paper is organized as follows. In Section 2, a new fragile watermarking scheme for 3D meshes is proposed in detail. The experiment results are given and discussed in Section 3. Finally, we draw a conclusion in Section 4.

2. A NEW APPROACH TO 3D MESHES AUTHENTICATION

We perform our watermarking process on meshes, which are considered as the “lowest common denominator” of surface representations. It is easy to convert other representations of 3D models to meshes. As described in [9], the mesh geometry can be denoted by a tuple (C, V) , where C is a simplicial complex specifying the connectivity of the mesh simplices (the adjacency of the vertices, edges, and faces), and $V = \{v_1, \dots, v_m\}$ is the set of vertex positions defining the shape of the mesh in V^3 .

2.1 Extending Dither Modulation to the Mesh Model

We aim to authenticate both geometry and topology of the mesh model, i.e., both connectivity and positions of the vertices need to be verified not having been modified. An implementation of quantization index modulation (QIM) [1] called dither modulation [2] is extended to the mesh model. The ratio of the defined distances is chosen as the embedding primitive, which is invariant to translation, rotation and uniformly scaling as shown below.

The watermarking process is based on the global characteristics of the original mesh, mainly including its centroid position and the largest dimension. The quantization step of dither modulation is chosen adaptively to the mesh geometry. Suppose $V = \{v_1, \dots, v_m\}$ is the set of vertex positions in V^3 , the position v_c of the mesh centroid is defined by

$$v_c = \frac{1}{m} \sum_{i=1}^m v_i. \quad (1)$$

The Euclidean distance d_i from a given vertex with the position v_i to the mesh centroid is given by

$$d_i = \sqrt{(v_{ix} - v_{cx})^2 + (v_{iy} - v_{cy})^2 + (v_{iz} - v_{cz})^2}, \quad (2)$$

where $\{v_{ix}, v_{iy}, v_{iz}\}$ is the vertex coordinate in V^3 . Using (2), we can find out the furthest vertex to the mesh centroid. Suppose v_d is the position of the furthest vertex, the corresponding distance D is denoted as

$$D = \sqrt{(v_{dx} - v_{cx})^2 + (v_{dy} - v_{cy})^2 + (v_{dz} - v_{cz})^2}. \quad (3)$$

And D is defined as the largest dimension of the mesh model. With the specified value of the parameter N , the quantization step S is chosen as

$$S = D/N. \quad (4)$$

The distance from a given face to the mesh centroid is defined as the Euclidean distance from the centroid of the face

to that of mesh geometry. Furthermore, the centroid position v_{ic} of a given face f_i with u edges is obtained by

$$v_{ic} = \frac{1}{u} \sum_{j=1}^u v_{ij}, \quad (5)$$

where $v_{ij}, j \in \{1, 2, \dots, u\}$ is the vertex position in the face f_i . The distance d_{fi} from the face f_i to the mesh centroid can be calculated using

$$d_{fi} = \sqrt{(v_{icx} - v_{cx})^2 + (v_{icy} - v_{cy})^2 + (v_{icz} - v_{cz})^2}. \quad (6)$$

Subsequently, the integer quotient Q_i and the remainder R_i are obtained by

$$Q_i = \lfloor d_{fi}/S \rfloor, \quad (7)$$

$$R_i = d_{fi} \% S. \quad (8)$$

To embed one bit value $w(i)$, we shift the centroid position of face f_i so that Q_i is an even value for the bit value 0, or an odd value for 1. In order to make $Q_i \% 2 = w(i)$ always hold, the distance d_{fi} is modulated by the following way

$$d'_{fi} = \begin{cases} Q_i \times S + S/2 & \text{if } Q_i \% 2 = w(i) \\ Q_i \times S - S/2 & \text{if } Q_i \% 2 = \overline{w(i)} \text{ \& } R_i < S/2 \\ Q_i \times S + 3S/2 & \text{if } Q_i \% 2 = \overline{w(i)} \text{ \& } R_i \geq S/2 \end{cases}, \quad (9)$$

where d'_{fi} is the modulated distance from the face f_i to the mesh centroid and $\overline{w(i)} = 1 - w(i)$. It is not hard to conclude that the modulated integer quotient

$$Q'_i = \lfloor d'_{fi}/S \rfloor = \begin{cases} Q_i & \text{if } Q_i \% 2 = w(i) \\ Q_i - 1 & \text{if } Q_i \% 2 = \overline{w(i)} \text{ \& } R_i < S/2 \\ Q_i + 1 & \text{if } Q_i \% 2 = \overline{w(i)} \text{ \& } R_i \geq S/2 \end{cases} \quad (10)$$

and $Q'_i \% 2 = w(i)$. The resulting d'_{fi} is used to adjust the position of the face centroid and only one vertex in the face is selected. Suppose v_{is} is the position of the selected vertex in the face f_i , which consists of u vertices with the centroid position v_{ic} , the adjusted vertex position would be

$$v'_{is} = [v_c + (v_{ic} - v_c) \times \frac{d'_{fi}}{d_{fi}}] \times u - \sum_{j=1, j \neq s}^u v_{ij}, \quad (11)$$

where v_{ij} is the vertex position in the face f_i .

The watermark information embedded in our method is invariant to translation, rotation and uniformly scaling because the ratio between the distance from the mesh centroid to each surface face and the quantization step, which is proportional to D , remains the same after the model is translated, rotated or uniformly scaled. Otherwise, if the mesh model is processed by other operations that change the ratios, the formula $Q'_i \% 2 = w(i)$ will not hold and the embedded watermark will be undermined. Since we need to detect a trivial modification on the mesh model, the integer value Q'_i should be sensitive to the distance from the mesh centroid to the surface face. It can be achieved by assigning N a large value to obtain a small quantization step S with respect to the precision of 3D data.

In [4], the topology of mesh geometry is described by the set of adjacent vertices whose indices are less than that of a given vertex plus the vertex itself. In contrast, the face indices of the mesh model is used to represent the connectivity of vertices in our method. If there is any change in

the mesh topology, such as mesh decimation or mesh resampling, the face indices will be modified so that the watermark hidden in the distances from the mesh faces to the mesh centroid cannot be correctly retrieved. Therefore, unauthorized modification on mesh topology can be detected.

2.2 The Encoding Process

This subsection elaborates on how to adjust the positions of the mesh faces and eventually move their centroid to the desired positions. Since a face shares edges and vertices with its neighbors, adjusting one face position also moves its neighbors. So only one vertex in the face is selected to adjust the face centroid to the desired position so that one watermark bit is embedded. After that, the face position can be fixed by fixing the coordinates of all its vertices. Therefore, after the adjustment, all vertices in the face should be marked to prevent their positions from being modified by the subsequent encoding operations. To successfully extract the embedded information, the centroid position and the largest dimension of the mesh should be maintained during the encoding process.

Algorithm 1: The Centroid-Distance-Ratio Encoding Algorithm

```

Calculate the position of the mesh centroid
Unmark all the vertices in the mesh model
foreach vertex in the mesh model do
    calculate the distance to the mesh centroid
    if the newly calculated distance is larger than any of
    the former then
        update the longest distance to the mesh centroid
        update the furthest vertex to the mesh centroid
mark the furthest vertex to the mesh centroid
obtain the quantization step by specifying the value of
N
scramble the face indices I using the key K to generate
I'
foreach face  $f_i$  indexed by  $I'$  do
    if there is at least one unmarked vertex in  $f_i$  then
        obtain the centroid position of  $f_i$ 
        calculate the distance  $d_{f_i}$  to the mesh centroid
        modulate  $d_{f_i}$  according to the bit value  $w(i)$  of
        the watermark
        adjust the position of one unmarked vertex in  $f_i$ 
        mark all vertices in  $f_i$ 
        if all the watermark information has been
        embedded then
            the embedding process is finished
        else
            go to the next face indexed by  $I'$ 
        else
            skip to the next face indexed by  $I'$ 
the centroid restoration process

```

The detailed procedure to embed the watermark information is shown in Algorithm 1. At first, all vertices of the original mesh are unmarked and the position of mesh centroid is obtained by (1). Then, the furthest vertex to the centroid is found using (2) and its corresponding distance to the mesh centroid is calculated using (3). Since the largest dimension D of the mesh model should not be changed, the furthest vertex is marked and its position will

not be modified. After that, the quantization step S is chosen by specifying the value of N in (4). To enhance the security of the watermarking scheme, we use the key K as the seed to scramble the face indices I to obtain the scrambled indices I' , which we will follow in the encoding process. Before one watermark bit is embedded by adjusting the distance from a face to the mesh centroid, all vertices in the face need to be checked and only the unmarked vertices are selected. If there is at least one unmarked vertex in a face f_i , the face is qualified to carry one bit information and one unmarked vertex is selected. The distance from the face to the mesh centroid is calculated by (6) and adjusted by (7-9) according to the value of the watermark bit. Since the largest dimension of the mesh model need to be maintained in the encoding process, if the modified distance exceeds it, twice of the quantization step should be subtracted from it so that the embedded value is kept. Then, the position of the selected vertex is adjusted using (11), whereby the face centroid is moved to the desired position. After the adjustment, all vertices in the face f_i will be marked. If there is no unmarked vertex in f_i , it implies that the face's candidate for embedding watermark information is not qualified and the checking mechanism will skip to the next face indexed by I' until all the watermark information is embedded.

The above embedding process inevitably introduces the distortion to the mesh geometry as some vertex coordinates are changed. However, the distortion can be limited to a predefined range because the elongate or the reduction of the distance is no more than twice of the quantization step in the proposed embedding algorithm. The distortion of the mesh geometry also change the position of the mesh centroid, although adjusting the vertex position may counteract each other. So not all faces can be used to embed the watermark information. Otherwise, the position of the original mesh centroid will be lost. A small portion of the vertices are needed to restore it after the embedding process. We refer to this process as the centroid restoration process, which modifies the coordinates of the unmarked vertices in the last faces indexed by I' to compensate the error introduced by the embedding process.

The centroid restoration process begins with the calculation of the introduced error during the embedding process by

$$E = \sum_{j=1}^m v'_j - \sum_{j=1}^m v_j, \quad (12)$$

where v_j and v'_j are the vertex positions before and after the embedding process, respectively. To ensure that the distance from the mesh centroid to the adjusted vertex will not exceed D , the positions of the unmarked vertices are adjusted in the following way:

Step 1: The admissible adjusting radius r_j of an unmarked vertex with the position v_j is calculated by

$$r_j = D - \sqrt{(v_{jx} - v_{cx})^2 + (v_{jy} - v_{cy})^2 + (v_{jz} - v_{cz})^2}; \quad (13)$$

Step 2: The value of r_j is used to weight the adjusting vector of each unmarked vertex. Supposing there are L unmarked vertices after the embedding process, the individual adjusting weight e_j can be obtained by

$$e_j = E \cdot \frac{r_j}{\sum_{k=1}^L r_k}. \quad (14)$$

Step 3: The corresponding adjusting weight is subtracted

from v_j to restore the position of the mesh centroid by

$$v'_j = v_j - e_j, \quad (15)$$

where v'_j represents the adjusted vertex position and v_j the original vertex position.

The encoding process ends as the position of the mesh centroid is restored.

2.3 The Decoding Process

In the decoding process, only the value of parameter N , the key K and the original watermark W are needed to authenticate the watermarked mesh. The detailed procedure is shown in Algorithm 2.

Algorithm 2: The Centroid-Distance-Ratio Decoding Algorithm

```

Calculate the position of the mesh centroid
Unmark all the vertices in the mesh model
foreach vertex in the mesh model do
  calculate the distance to the mesh centroid
  if the newly obtained distance is larger than any of
  the former then
    update the longest distance to the mesh centroid
    update the furthest vertex to the mesh centroid
Mark the furthest vertex to the mesh centroid
Obtain the quantization step  $S'$  by the given value of  $N$ 
Scramble the face indices  $I$  using the given key  $K$  to
generate  $I'$ 
foreach face  $f'_i$  indexed by  $I'$  do
  if there is at least one unmarked vertex in  $f_i$  then
    obtain the centroid position of  $f'_i$ 
    calculate the distance  $D_{f_i}$  to the mesh centroid
    calculate the integer quotient  $Q'_i = D_{f_i}/S'$ 
    extract the bit value by  $w(i) = Q'_i \% 2$ 
    mark all vertices in  $f_i$ 
    if all the watermark information has been
    retrieved then
      the extraction process is finished
    else
      go to the next face indexed by  $I'$ 
    else
      skip to the next face indexed by  $I'$ 
the cross-correlation calculation

```

Similar to the encoding process, all the vertices in the mesh model are unmarked initially. The position v'_c of the mesh centroid can be obtained by (1). Then, the furthest vertex to the centroid is found using (2) and its corresponding distance D' to the mesh centroid is calculated by (3). Since the furthest vertex is marked before the embedding process, it should also be marked before the extraction process. With the provided parameter N , the quantization step S' is calculated by

$$S' = D'/N. \quad (16)$$

Then the face indices I is scrambled using the key K as the seed to produce the scrambled indices I' . Before calculating the centroid position of a given face, the marks of its vertices should be checked. If there is at least one unmarked vertex in a face f'_i , the face will be qualified to extract the embedded bit and its centroid position v'_{ic} will be calculated using (5).

Then, the modulated distance d'_{f_i} from the face f'_i to the mesh centroid is calculated by

$$d'_{f_i} = \sqrt{(v'_{icx} - v'_{cx})^2 + (v'_{icy} - v'_{cy})^2 + (v'_{icz} - v'_{cz})^2}, \quad (17)$$

And the integer quotient Q'_i is obtained by

$$Q'_i = \lfloor d'_{f_i}/S' \rfloor. \quad (18)$$

Eventually, the bit information $w'(i)$ is extracted by

$$w'(i) = Q'_i \% 2. \quad (19)$$

At the end of the extraction operation, all the vertices in the face f'_i are marked. If there is no unmarked vertex in f'_i , no information is extracted and the decoding process will automatically skip to the next face indexed by I' . Since we know the original watermark W , the watermark extraction operation will cease once the number of the extracted bits matches the number of the embedded bits.

After the extraction process, the extracted watermark W' is compared with the original watermark W using the following cross-correlation function to measure the their correlation. Supposing their lengths are both identical to K , the normalized cross-correlation value NC between the original and the extracted watermarks is calculated by

$$NC = \frac{1}{K} \sum_{i=1}^K I(w'(i), w(i)) \quad (20)$$

where

$$I(w'(i), w(i)) = \begin{cases} 1 & \text{if } w'(i) = w(i) \\ -1 & \text{otherwise} \end{cases}. \quad (21)$$

If the watermarked mesh geometry is intact, the NC value will be 1; otherwise, it will be less than 1. We claim the mesh geometry as being tampered if the resulting NC value from (20) is less than 1.

2.4 The Enhanced Centroid-Distance-Ratio Algorithm

In our proposed method, the embedded watermark is invariant to translation, rotation and uniformly scaling, but sensitive to other processing. The encoding and decoding algorithms can also be enhanced to detect those operations by using a reference position p_r .

In the encoding process, besides one unmarked vertex in f_i is selected to modulate the distance from the face f_i to the mesh centroid, another vertex in f_i is chosen as the reference vertex. Then the distance from the reference vertex to the reference position p_r is calculated by

$$|v_{ir} - p_r| = \sqrt{(v_{irx} - p_{rx})^2 + (v_{iry} - p_{ry})^2 + (v_{irz} - p_{rz})^2}, \quad (22)$$

where v_{ir} is the position of the reference vertex. Then R_{ir} is obtained by

$$R_{ir} = (|v_{ir} - p_r| \% S - S/2)/2. \quad (23)$$

After that, we adopt (9) in the following way

$$d'_{f_i} = \begin{cases} Q_i \times S + S/2 + R_{ir} & \text{if } Q_i \% 2 = \underline{w(i)} \\ Q_i \times S - S/2 + R_{ir} & \text{if } Q_i \% 2 = \overline{w(i)} \ \& \ R_i < S/2 \\ Q_i \times S + 3S/2 + R_{ir} & \text{if } Q_i \% 2 = \overline{w(i)} \ \& \ R_i \geq S/2 \end{cases}. \quad (24)$$

Because $0 \leq |v_{ir} - p_r| \% S < S$, $(|v_{ir} - p_r| \% S - S/2)/2$ is within the interval $[-S/4, S/4]$. Therefore, the adding

of R_{ir} in (23) will not interfere the watermark information hidden in the distance d'_{fi} . It can be derived from (24) that $R_{ir} = d'_{fi} - Q'_i \times S - S/2$ and $Q'_i \% 2 = w(i)$, where the integer quotient $Q'_i = \lfloor d'_{fi}/S \rfloor$.

In the decoding process, the reference position p_r need to be provided. Firstly, the modulated distance d'_{fi} is calculated using (17). Then the integer quotient Q'_i can be obtained by (18) and the bit value $w'(i)$ is retrieved by (19). The distance from the reference vertex in f_i to the reference position p_r is calculated by (22) and R_{ir} is obtained by (23). After that, the difference between R_{ir} and $d'_{fi} - Q'_i \times S - S/2$ is calculated. For each face that carries the watermark information, if $|d'_{fi} - Q'_i \times S - S/2 - R_{ir}| < \epsilon$ and the extracted bit $w'(i)$ equals to the original watermark bit $w(i)$, the mesh model is considered as not having been modified.

If the watermarked mesh model is processed by translation, rotation and uniformly scaling operations, the vertex positions will be modified and the distance from the reference position to the reference vertex in f_i will change, as well as the ratio between R_{ir} and the quantization step S . However, the ratio between the distance d'_{fi} and S , as well as the ratio between $d'_{fi} - Q'_i \times S - S/2$ and S , remains the same. If a relatively small value, $S/8$ for example, is assigned to ϵ , then $|d'_{fi} - Q'_i \times S - S/2 - R_{ir}| > \epsilon$. translation, rotation and uniformly scaling operations will be identified if R_{ir} does not match $d'_{fi} - Q'_i \times S - S/2$ while the extracted watermark is the same as the original one. Although the reference position p_r can be arbitrarily chosen, it should be chosen nearby the mesh centroid to make the ratio between R_{ir} and S more sensitive to those operations.

In case that the watermarked mesh model is processed by other geometrical modification, there exists the face f_i from which the distance d'_{fi} to the mesh centroid is altered. So the ratio between $d'_{fi} - Q'_i \times S - S/2$ and S is changed. However, the ratio between R_{ir} and S is not certain to vary, depending on whether the reference vertex in f_i is moved. Even if the ratio between R_{ir} and S is also changed, $|d'_{fi} - Q'_i \times S - S/2 - R_{ir}|$ will probably exceed ϵ . Therefore, besides the embedded watermark, the enhanced method can detect those operations other than translation, rotation and uniformly scaling that have been applied to the watermarked mesh model by comparing R_{ir} with $d'_{fi} - Q'_i \times S - S/2$.

Suppose there are K faces used to hide the watermark information, we need to compare R_{ir} with $d'_{fi} - Q'_i \times S - S/2$ for K times in authenticating the watermarked mesh model. To estimate the strength of tampering, the normalized matching number NM between R_{ir} and $d'_{fi} - Q'_i \times S - S/2$ is obtained by

$$NM = \frac{1}{K} \sum_{i=1}^K m(i) \quad (25)$$

where

$$m(i) = \begin{cases} 1 & \text{if } |d'_{fi} - Q'_i \times S - S/2 - R_{ir}| < \epsilon \\ 0 & \text{otherwise} \end{cases} \quad (26)$$

The mesh model will be considered as being tampered either NC in (20) or NM in (25) is less than 1.

3. EXPERIMENT RESULTS

We have investigated the proposed approach on several mesh models listed in Table 1 and a 2D binary image is chosen as the watermark, which can also be a hashed value.

Table 1: The Mesh Models Used In The Experiments

Models	Vertices	Faces	Capacity(bits)
dog	7158	13176	5552
wolf	7232	13992	5953
raptor	8171	14568	7544
horse	9988	18363	7650
cat	10361	19098	8035
lion	16652	32096	14460

In the case that translation, rotation and uniformly scaling operations need to be detected, the Enhanced Centroid-Distance-Ratio Algorithm is preferred. The original mesh model “lion” and its watermarked version are shown in Figure 1(a) and Figure 1(b).

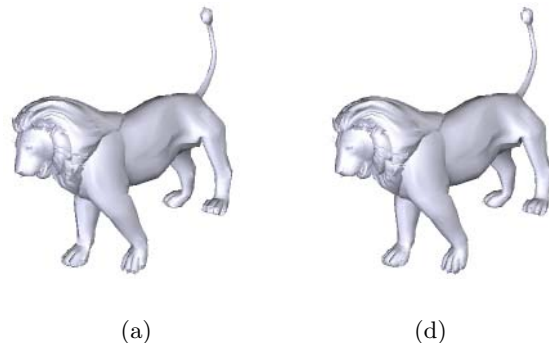


Figure 1: (a). The original mesh model “lion”; (b). The watermarked mesh model using the Enhanced Centroid-Distance-Ratio Algorithm

The capacities of the listed models are also shown in Table 1, provided that 0.25% vertices are used in the centroid restoration process. In our proposed method, the capacity of the mesh model to carry the watermark information is dependent on the vertex number. The bit number that can be hidden in the mesh model cannot exceed the vertex number, since adjusting one vertex position hides one bit information and there always exist vertices whose positions have not been adjusted during the watermark embedding process. We wish to hide as many watermark bits as possible in the mesh model so that the embedded watermark is more sensitive to unauthorized modifications.

To evaluate the imperceptibility of the proposed algorithm, the Hausdorff Distance between the original mesh model and the watermarked mesh model normalized by D was calculated to measure the distortion introduced by the encoding process as in [10], upon the fact that the mesh topology has not been changed during the watermarking process. Figure 2 shows the normalized Hausdorff Distance subject to the parameter of N , given that the percent of vertices used in the centroid restoration process is about 0.25%. It can be seen that the introduced distortion on the mesh model decreases as the parameter N increases.

The global characteristics such as the centroid position

Table 2: The NC between the original and extracted watermarks after some types of operations ($N = 100000$)

Models	translation, rotation, uniformly scaling	moving two vertices oppositely	modifying one vertex position	reducing one face	adding 0.001% noise	extracting without the key
dog	1.0000	0.9985	-0.2808	0.2301	-0.0064	0.0218
wolf	1.0000	0.9983	-0.2179	0.3392	-0.0038	0.0331
raptor	1.0000	0.9989	-0.2927	0.2287	0.0043	0.0335
horse	1.0000	0.9989	0.0652	0.3126	-0.0027	0.0333
cat	1.0000	0.9992	-0.1454	0.3040	-0.0045	0.0059
lion	1.0000	0.9991	-0.0496	0.1956	0.0001	0.0266

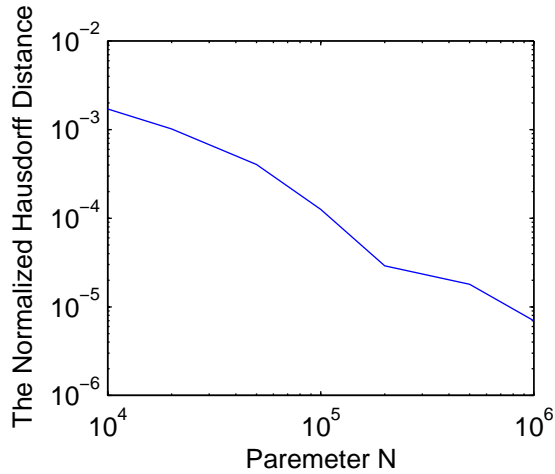


Figure 2: The normalized Hausdorff distance subject to the parameter N

and the largest dimension of the mesh model are used in the watermarking algorithm. If these global characteristics are slightly altered, the watermark information will be dramatically changed and the modifications on the watermarked mesh model can not be located. However, it is easy to locate the tampering if it is limited to a local area and has little impact on the global characteristics. Since we used a 2D binary image as the watermark, the impact of trivial modifications can be visualized in the extracted watermark image while severe modifications make it meaningless. With the extracted watermark, we can detect the unauthorized modifications and estimate the strength of tampering, if any.

Using the Enhanced Centroid-Distance-Ratio Algorithm, the watermarked mesh models went through translation, rotation and uniformly scaling operations, changing the positions of two vertices oppositely (respectively by adding the vectors $\{2S, 2S, 2S\}$ and $\{-2S, -2S, -2S\}$), modifying one vertex position by adding the vector $\{\frac{mD}{2N}, \frac{mD}{2N}, \frac{mD}{2N}\}$, reducing one face from the mesh, and adding the noise signal $\{n_x, n_y, n_z\}$ to all the vertex positions with n_x, n_y and n_z uniformly distributed within the interval $[-S, S]$, respectively. The mesh models after these operations are also listed

in Figure 3(a)-(e). The watermarks are extracted from the processed mesh models with and without the key K and the normalized cross-correlation value NC between the extracted and the original watermarks are calculated using (20). To detect translation, rotation and uniformly scaling operations, the normalized matching number NM between R_{ir} and $d'_{fi} - Q'_i \times S - S/2$ is also calculated. The resulting values are listed in the Table 2 and Table 3, respectively.

4. CONCLUSIONS

In this paper, we have proposed a new fragile watermarking of 3D mesh models for authentication purpose. The watermarking process is conducted in spatial domain and applies to all the mesh models without any restriction. The experimental results have demonstrated that the proposed method is able to adaptively embed a considerable amount of information into the mesh model. The embedded watermark can be blindly extracted from the watermarked mesh model with some priori knowledge and used to detect the unauthorized tampering with the watermarked mesh.

In our method, the distortion introduced by the encoding process is quite small and can be adjusted by assigned the parameter N a proper value, providing a trade-off between imperceptibility and the precision of 3D data. Compared with previous works, the embedded watermark by our approach is invariant to translation, rotation and uniformly scaling, but sensitive to other operations. Furthermore, the proposed method can also be enhanced to detect those operations besides other processing that might have been applied to the watermarked mesh. Therefore, unauthorized modifications on the mesh model can be detected and classified.

5. ACKNOWLEDGMENTS

This work was supported by the Research Grant Council of Hong Kong SAR under Project HKBU 2156/04E. Many thanks to the reviewers for their insightful comments and suggestions which improved the quality of the paper.

For the use of the 3D models, we would like to thank the web source <http://www.cs.unc.edu/isenburg/ac/models/src/> at the Department of Computer Science, the University of North Carolina at Chapel Hill, USA.

Table 3: The normalized matching number NM after some types of operations ($N = 100000$)

Models	translation	rotation	uniformly scaling	moving two vertices oppositely	modifying one vertex position	reducing one face	adding 0.001% noise
dog	0.4304	0.4449	0.4462	0.9994	0.0732	0.7752	0.2533
wolf	0.4292	0.4296	0.4330	0.9994	0.0891	0.7521	0.2413
raptor	0.4486	0.4428	0.4350	0.9996	0.0975	0.7480	0.2482
horse	0.4457	0.4360	0.4492	0.9996	0.2742	0.7369	0.2475
cat	0.4428	0.4405	0.4308	0.9997	0.1461	0.7820	0.2568
lion	0.4419	0.4418	0.4329	0.9996	0.0740	0.8066	0.2490

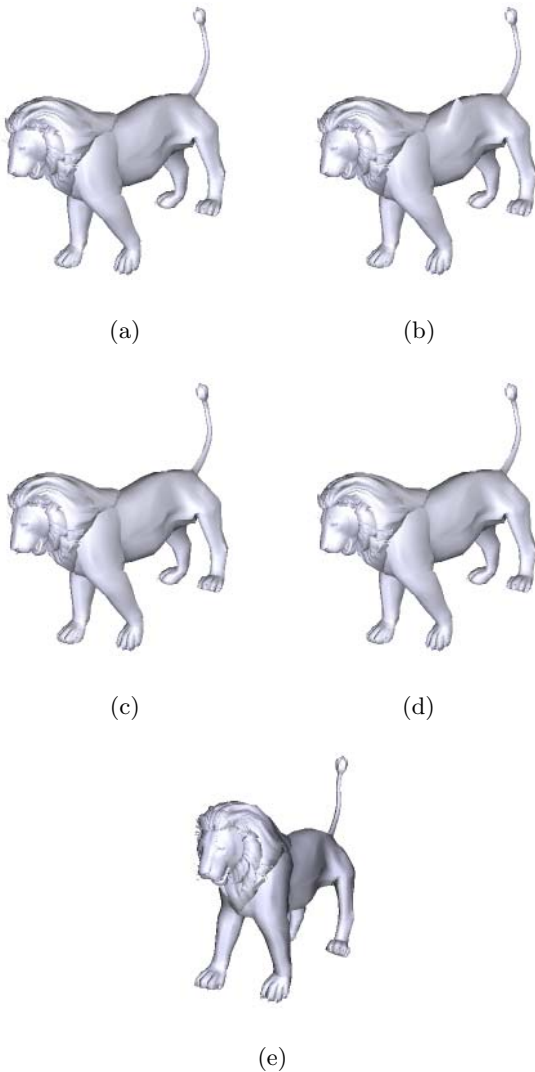


Figure 3: The watermarked mesh models after (a). moving two vertices moved oppositely; (b). one vertex position modified; (c). reducing one face; (d). adding noise (e). translation, rotation and uniformly scaling operations

6. REFERENCES

- [1] B. Chen and G.W.Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 1423-1443, May 2001.
- [2] B. Chen and G.W.Wornell, "Digital watermarking and information embedding using dither modulation," *IEEE Second Workshop on Multimedia Signal Processing*, pp. 273-278, 1998.
- [3] Z. Q. Yu, H. H. S. Ip and L. F. Kwork, "A robust watermarking scheme for 3D triangle mesh models," *Pattern Recognition*, Vol. 36, pp. 2603-2614(12), 2003.
- [4] M. M. Yeung and B.-L. Yeo, "Fragile watermarking of three dimensional objects," *Proc. 1998 Int'l Conf. Image Processing, ICIP98*, volume 2, pp. 442-446. IEEE Computer Society, 1998.
- [5] B.-L. Yeo and M. M. Yeung, "Watermarking 3-D objects for verification," *IEEE Comput. Graph. Applicat.*, pp. 36-45, Jan./Feb. 1999.
- [6] C. Fornaro and A. Sanna, "Public key watermarking for authentication of CSG models," *Computer-Aided Design*, vol. 32, pp.727-735, 2000.
- [7] Hsueh-Yi Lin, Hong-yuan Mark Liao, Chun-Shien Lu and Ja-Chen Lin, "Fragile watermarking for authenticating 3-D polygonal meshes," *Proc. 16th IPPR Conf on CVGIP*, pp. 298-304, 2003
- [8] O. Benedens and C. Busch, "Toward blind detection of robust watermarks in polygonal models," *Proc. EUROGRAPHICS Comput. Graph. Forum*, vol. 19, pp. C199-C208, 2000.
- [9] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald and W. Stuetzle, "Mesh optimization," *Computer Graphics (SIGGRAPH '93 Proceedings)*, pp.19-26, 1993.
- [10] F. Cayre and B. Macq, "Data hiding on 3-D triangle meshes," *IEEE Trans. Signal. Processing*, vol. 51, pp. 939-949 (4), 2003.