Tamperproof Watermarking of 3D Models using Hausdorff Distance

Mukesh Motwani[#1], Balaji Sridharan[*2], Rakhi Motwani[#1], and Frederick C. Harris, Jr.[#1]

[#1]Univerity of Nevada, Reno USA
[1]mukesh@cse.unr.edu
[3]fredh@cse.unr.edu
[*2]Vishwakarma Institute of Technology, Pune India
[2]balaji.sridharan@ieee.org

Abstract – **This paper describes a novel algorithm for tamperproof watermarking of 3D models. Fragile watermarking is used to detect any kind of tamper i.e. unauthorized modifications in the model. The best and the simplest way to do this is by inserting a watermark at each and every vertex of the model. This poses as a challenge as insertion of watermark in every vertex can cause perceptible distortion and inserting such a watermark is computationally expensive. The challenge of perceptible distortion is overcome by using a measure that controls perceptible distortional called the hausdorff distance. Thus, the objective of the Genetic Algorithm is to minimize the hausdorff distance between the 2 ring neighbourhood of the original and the watermarked vertex. The other challenge of time complexity is overcome by running the Genetic Algorithm for just 20 generations and causing it to converge prematurely. This significantly reduces the computational cost. The experimental results indicate that the algorithm effectively detects any distortion in model.**

***Keywords*— fragile watermarking, genetic algorithms, 3D mesh model, SNR, 2 ring neighbourhood.**

I. Introduction

There has been a significant development in the 3D digital content in the past few years. Creation of 3D models is a very difficult task and the designers share and exhibit this artwork. However, the duplication and unauthorized modification of these models has made 3D watermarking a necessity. There are 3 types of 3D watermarking. *Robust Watermarking* is used to claim ownership rights and *Fragile Watermarking* is used to prevent any kind of unauthorized modification. Thus, it for simple and efficient detection of unauthorized modification, it is imperative that each and every vertex of the model must be watermarked. This poses two challenges, the first one being the challenge to maintain the imperceptibility and the second being the high computational cost. This paper successfully solves that challenge by using a Hausdorff Distance parameter and the concept of premature convergence of the GA. [1]

A comprehensive survey of 3D model watermarking techniques has been done by [2]. Genetic Algorithms have been rarely used to watermark 3D models; although they have been widely is image and video watermarking. In [3], the GA uses Peak Signal to Noise Ratio (PSNR) to avoid perceptible distortion and Normalized Cross Correlation (NCC) for the robustness of the watermark. The watermark has been inserted by modifying the DCT (Discreet Cosine Transform) coefficients. [4] also employs the same technique. Hausdorff distance has been used for a variety of pattern recognition applications. In [5] a new weighting function of Hausdorff distance measure termed Edge Eigenface Weighted Hausdorff Distance (EEWHD) has been proposed for face recognition. Hausdorff distances are generally used to find out the degree of mismatch between two sets of points. [6] considers these sets of points as “model” set and “image” set. A method for comparing the images using Hausdorff distances has been proposed. The paper provides efficient algorithms for computing the Hausdorff distance between all possible relative positions of a binary image and a model. An approach for using hausdorff distance as a measure of degree of match rather than mismatch is explained in [7]. This similarity information can especially better the discriminating capability of a pattern recognition system for similar objects. The use of Hausdorff distance as an error measuring parameter has been explained in MESH [8]. Hausdorff Distance has been previously used in image watermarking [9]. In this paper, a general framework for image content authentication using salient feature points was generated and then compared, while extraction using hausdorff distance. Besides this, Hausdorff distance has been used for correcting the geometric distortions by comparing the feature sets before and after experience distortion in [10]

Evolutionary algorithms (EAs) are algorithms that use the concepts of biological phenomena to solve optimization problems. Various types of EAs are particle swarm optimization, ant colony optimization, genetic algorithms and genetic programming. Genetic Algorithms [11] use biological evolution and survival of the fittest to solve computational problems. The solution begins with the creation of the initial population also called chromosomes. These chromosomes are then passed through the fitness function or the objective function. The fittest are selected to reproduce. The reproduction process is three fold. Elitism carries the best in the generation to the next generation. During crossover there is a random exchange of genes between the chromosomes to form an offspring. Mutation randomly flips one of the genes of the chromosome to give a new offspring. This new generation is then again passed through the fitness function and the process continues till stopping criteria is reached

II. Background

A. *3D Model Representation*

A 3D model consists of a 3D mesh. A 3D mesh consists of a vertex list and a face list. Vertex list gives the information about the orientation of the vertices in space and the face list gives the information about how these vertices are connected to each other. Thus, the vertex list gives the (x, y, z) Cartesian coordinates of the vertices that constitute the vertex list. The Cartesian coordinates also represent the chromosome.

One ring neighborhood of the vertex V is defined as the collection of surfaces formed by that vertex with its neighbors.

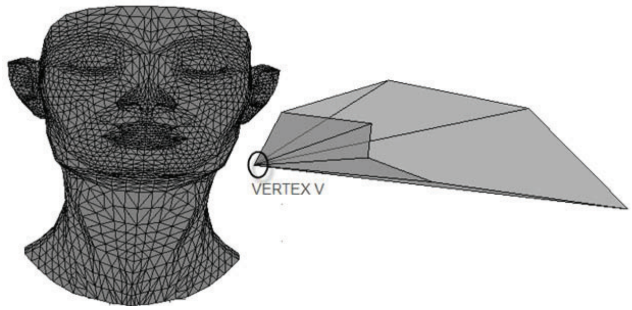

Fig 1. *Nefertiti* triangular mesh model and one ring neighborhood of vertex index 26. The vertex 'V' is shown by a circle around it.

Similarly, a 2 ring neighborhood of the vertex V is defined as the collection of the one ring neighborhoods of all the vertices in the one ring neighborhood of the vertex V.

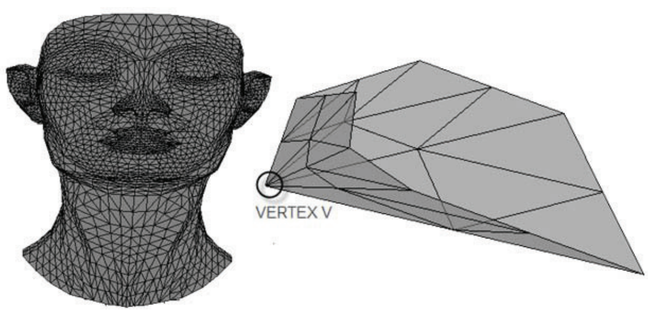

Fig. 2. The two ring neighborhood of the vertex index 26 of *Nefertiti*

B. *Hausdorff Distance*

For given sets of points, the hausdorff distance is defined as the maximum distance of a set to the nearest point in the other set. Thus, hausdorff distance can be represented as a maximin function as shown in equation 1

$$h(a,b) = \max\{min(d(a,b))\} \qquad (1)$$

Here, '*a*' and '*b*' represent the sets of points between whom the hausdorff distance is measured. In the algorithm, '*a*' is the vertex list formed by the two ring neighbourhood of the vertex V and '*b*' is the vertex list formed by the two ring neighbourhood of V', which is the position of V on watermark.

It should, however be noted that $h(a,b) \neq h(b,a)$.
Thus, the symmetrical hausdorff distance is defined as

$$H(a,b) = \max\{h(a,b)\ h(b,a)\} \qquad (2)$$

Algorithm to find out the hausdorff distance: -

1. h = 0
2. for every point a_i of set A,
    - 2.1 shortest = Inf ;
    - 2.2 for every point b_j of set B
        - $d_{ij} = d(a_i, b_j)$
        - if d_{ij} < shortest then
            - shortest = d_{ij}
    - 2.3 if shortest > h then
        - h = shortest

III. Proposed Approach

A. *Fitness Function*

For efficient fragile watermarking, it is necessary to modify all the vertices in the model. While doing so, great care has to be taken to avoid causing any distortion in the model.
Symmetrical hausdorff distance is used as an approach to find the degree of mismatch between two polygons or broadly, 2 sets of points in space. In this paper we have used Hausdorff Distance to find the degree of mismatch between the 2 ring neighborhoods of the original vertex V and the watermarked vertex V'. This degree of mismatch is reduced by minimizing the hausdorff distance and thus maintaining no perceptible distortion.

Thus the Hausdorff distance can be found out using [8] and [12].

$$Fitness\ Function = H(v,v') = \max\{\mathrm{h}(v,v')\ \mathrm{h}(v',v)\} \qquad (3)$$

B. *Genetic Algorithm parameters*

TABLE I – Genetic Algorithm Parameters

Population Size	100	
No. of generations	20 generations	
Initial range	Upper bound	Max value of the coordinates of 2-ring
	Lower bound	Min value of the coordinates of 2-ring
Creation Function	Uniform	
Selection Operator	Stochastic uniform	
Crossover Operator	Scattered Crossover	
Crossover Rate	0.8	
Mutation Operator	Uniform Mutation	

Mutation Rate	0.001
Elite count	2
Fitness Threshold Value	10^{-09}

C. *Watermark generation and insertion*

The algorithm for watermark insertion: -
for i=1:number of vertices

1. find the 2ring neighbourhood of the vertex index 'i'
2. Generate the initial population using uniform creation function
3. Find the fitness values using hausdorff distance
4. Sort the fitness values.
5. Use Selection operator to find the fittest individuals
6. Carry forward the Elite individuals
7. Obtain the offspring by crossover
8. Obtain the offspring by mutation
9. Create the new generation
10. If stopping criteria met return V'
    Else go to 3.

end for.

2ring neighbourhood is used here as the level of accuracy increases with a two ring neighbourhood since the comparison is done with a larger number of vertices compared to the one ring. Thus, we find the degree of mismatch of the 2ring neighbourhood of the original vertex V and the new position of this vertex V'.
The operators used here are:
Uniform Creation fnction is used for generating the initial population. This generates an uniformly distributed population.
Stochastic Selection as this type of selection works on the principle of survival of the fittest.
Scattered Crossover was used as it adds to the degree of randomness in the newer generation since the point of crossover and the number of crossover points is not predefined.
Uniform Mutation is used as this operator replaces a vertex of the chromosome by a random value within the range.

IV. EXPERIMENTAL RESULTS

A. *Watermarking the models.*

The models used for watermarking were Nefertiti, Venus, Mushroom and Pipes. The algorithm was run for 20 generations with a population size of 100. The algorithm was run by using both one ring as well as two ring neighborhoods.

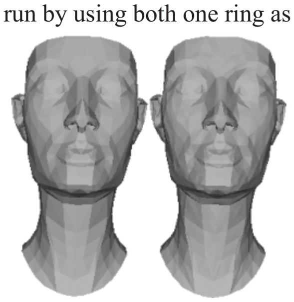

Fig 3a) Nefertiti

Fig 3b) Venus

Fig 3c) Mushroom

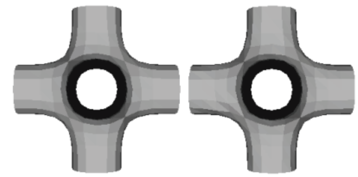

Fig 3d) Pipes

Fig 5. The models to the left are original models and the models to the right are watermarked models using 2 ring neighborhood.

TABLE II

Model	Nefertiti	Venus	Mushroom	Pipes
No. of vertices in the original model	2496	711	869	428
No. of vertices watermarked	2496	711	869	428
SNR (Signal to noise ratio) using 2 ring neighborhood	113.129 db	101.565 db	113.83 db	123.42db
Hausdorff Distance using 2 ring neighborhood	0.000613	0.012742	0.005304	0.00214
SNR (Signal to noise ratio) using 1 ring neighborhood	101.14 db	90.28 db	91.35db	113.21 db
Hausdorff Distance using 1 ring neighborhood	0.002255	0.01885	0.008758	0.014421

As the table 2 shows when watermark is generated and inserted using a 2 ring neighborhood, both the SNR (Signal to noise ratio) and the hausdorff distance are found to be better. Thus, it is observed that watermarking using 2 ring neighborhood gives better results. The table also shows that the algorithm works efficiently for low as well as high resolution models. Figure 3 shows that there is no perceptible distortion between the original and the watermarked model.

B. *Analysis of flat surfaces*

The Genetic Algorithm has to generate a watermark in 3D models that have flat surfaces as well. This is difficult because the vertex V can move only on a flat surface on a 2D plane. This is ensured by the fitness function.
Figure 4 shows a cube where the mesh structure is seen to be distorted but the model remains undistorted.

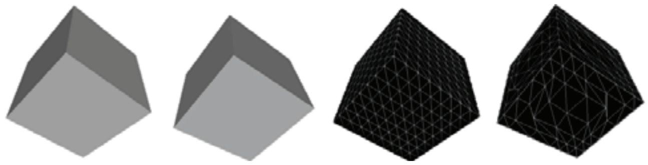

Fig.4. Cube (from left to right: -Original Model, Watermarked Model, Mesh structure original model, Mesh structure of the watermarked model)

Thus, we can conclude from figure 4 that the algorithm works efficiently for models with flat surfaces. The distorted mesh shows that there has been a significant insertion of watermark and the perceptibility of the model hasn't been compromised.
Along with the flat surfaces, the algorithm also efficiently watermarks the cylindrical surfaces. This is exhibited in figure 5.
Mechanical (figure 5) is a combination of both cylindrical as well as flat surfaces. Figure 5 shows that Genetic Algorithm watermarks the Mechanical model without any perceptible distortion. Thus, the GA works for flat as well as cylindrical surfaces.

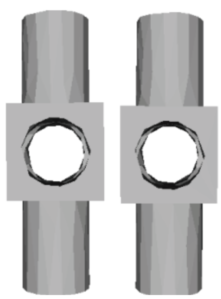

Fig.5. Mechanical (Model to the left is original and to the right is watermarked)

C. *Attacks on the model*

The attacks for fragile watermarking of 3D model differ from the attacks for robust watermarking. The attacks on the fragile watermarked 3D model deal only with unauthorized modification of any region in the model.

Algorithm for detection of unauthorized data modification: -

1. Find the correlation between the suspicious and the watermarked model
2. If correlation = 100%

   Model has not been tampered

   else

   compare the 2 models to detect the region of tamper

Correlation is found using equations 4 and 5.

$$Amount\ of\ Correlation = \frac{w + w' * correlation}{W} \tag{4}$$

$$correlation = \sum_{i=1}^{N} \frac{A(i).A'(i)}{|A(i)| * |A'(i)|} \tag{5}$$

Where

$$A(i) = A_x(i) + A_y(j) + A_z(k)$$

$$A(i) = A_x'(i) + A_y'(j) + A_z'(k)$$

Where

w = no. of vertices not attacked.
w' = no. of vertices attacked.
W = total no. of vertices in the watermarked model.
$A_x(i)$, $A_y(i)$, $A_z(i)$ are the x,y,z co-ordinates of the i^{th} vertex in the attacked model.
$A_x'(i)$, $A_y'(i)$, $A_z'(i)$ are the x,y,z co-ordinates of the corresponding i^{th} vertex in the watermarked model.

The watermark is not affected by attacks such as translation, rotation and scaling as shown in. Fig 6 and Fig 7. Correlation of 100% is obtained for affine transformations. This is because the model has been normalized before watermarking.
The other type of attacks are deformation or cropping attacks, where the vertices are deformed as shown in Fig 8. In such a case, the algorithm should be able to find out the location at which the deformation or cropping has taken place as shown in Fig 8. Besides this, the models were also subjected to multiple deformations as shown in the figure 9. The figures were generated using [13].

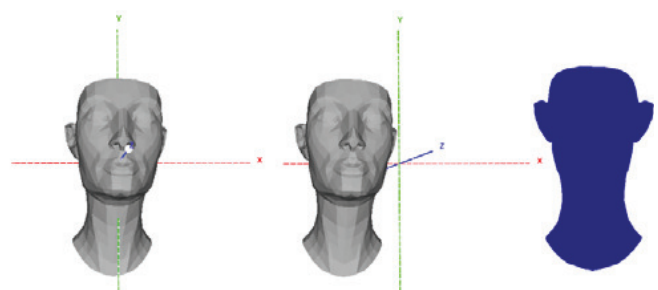

Fig.6. Translation on *Nefertiti* Model

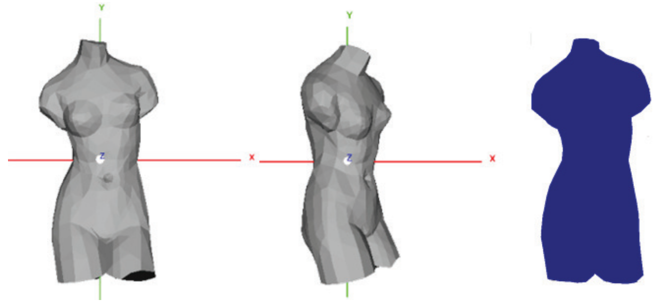

Fig.7. Rotation on *Venus* model

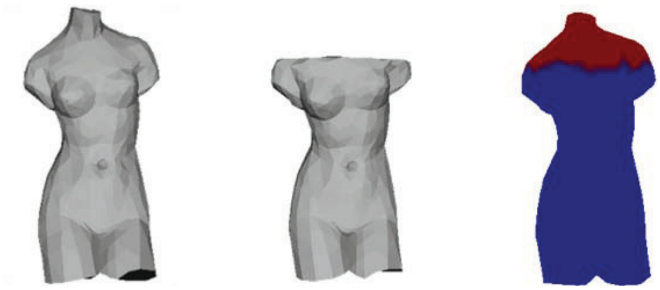

Fig.8. Modification in *Venus* model

Fig.9. Multiple modifications in *Nefertiti* model.

In figures 6-9, the model to the left is the watermarked model, the model at the centre is the deformed model and the model to the right is the detection of area of deformation.

As shown in figures 6 and 7, the translating or rotating the model does not affect the mesh structure of the model and thus, there is no unauthorized modification. This is shown by the absence of red patch.

Figure 8 shows that the upper part of the *Venus* model has been modified. The area of tamper is shown by the red patch.

Figure 9 shows that the algorithm can also detect multiple modifications in the model. The red patches in the *Nefertiti* model show the areas of deformation

V. CONCLUSION

In this paper, a fragile watermarking technique using Genetic Algorithms has been proposed. The algorithm generates and embeds a watermark in each and every vertex of the model. The use of hausdorff Distance as the fitness function prevents any perceptible distortion in the model. Also, though Genetic Algorithms are known to be slow, the approach used here reduces the computational costs significantly. The average time for watermarking each vertex was approximately 1.5 seconds. Also, for better accuracy, a 2 ring vertex is used. This helps in determining the degree of mismatch relative to a larger number of vertices, which enables to decrease the perceptible distortion. This is shown by the Hausdorff distance and the SNR values in table 2. The algorithm has been analyzed for flat and cylindrical surfaces. It was shown that the mesh structure in case of flat surfaces is distorted though the model is not, which indicated insertion of significant amount of watermark. The algorithm was tested for the unauthorized modification attack. It was shown that the algorithm detects the region of tamper effectively.

REFERENCES

[1] Motwani, M.C., Balaji S, Motwani, R.C.,"Tamper proof watermarking of 3D mdoels", *International Conference on Signal Acquisition and Processing, Second International Conference on,* vol. 1,no, pp210-215 Feb. 2010

[2] Jean-Luc Dugelay , Atilla Baskurt, Mohamed Daoudi, "3D Object Processing: Compression, Indexing and Watermarking, " Wiley, 2008

[3] Yongqiang, Chen; Lihua, Peng, "Optimal Image Watermark Using Genetic Algorithm and Synergetic Neural Network," *Intelligent Computation Technology and Automation, 2009. ICICTA '09. Second International Conference on* , vol.3, no., pp.209-212, 10-11 Oct. 2009

[4] ElShafie, D.R.; Kharma, N.; Ward, R., "Parameter optimization of an embedded watermark using a genetic algorithm," *Communications, Control and Signal Processing, 2008. ISCCSP 2008. 3rd International Symposium on* , vol., no., pp.1263-1267, 12-14 March 2008

[5] Huachun Tan; Yu-Jin Zhang, "Computing Eigenface from Edge Images for Face Recognition Based on Hausdorff Distance," *Image and Graphics, 2007. ICIG 2007. Fourth International Conference on* , vol., no., pp.639-644, 22-24 Aug. 2007

[6] Huttenlocher, D.P.; Klanderman, G.A.; Rucklidge, W.J., "Comparing images using the Hausdorff distance," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , vol.15, no.9, pp.850-863, Sep 1993

[7] Yuankui Hu; Zengfu Wang, "A Similarity Measure Based on Hausdorff Distance for Human Face Recognition," *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on* , vol.3, no., pp.1131-1134

[8] Aspert, N.; Santa-Cruz, D.; Ebrahimi, T., "MESH: measuring errors between surfaces using the Hausdorff distance," *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on* , vol.1, no., pp. 705-708 vol.1, 2002

[9] Monga, V.; Vats, D.; Evans, B.L., "Image Authentication Under Geometric Attacks Via Structure Matching," *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on* , vol., no., pp.229-232, 6-6 July 2005

[10] Xiaoli Zhang; Lv Xin, "A Novel Watermarking Algorithm Resist to Geometrical Attacks," *Electronic Commerce and Business Intelligence, 2009. ECBI 2009. International Conference on* , vol., no., pp.503-506, 6-7 June 2009

[11] Mitchell Melanie, "An Introduction to Genetic Algorithms," The MIT Press, 1999.

[12] Bennour, J.; Dugelay, J.-L., "Toward a 3D watermarking benchmark," *Multimedia Signal Processing, 2007. MMSP 2007. IEEE 9th Workshop on* , vol., no., pp.369-372, 1-3 Oct. 2007

[13] G. Peyré: "A Toolbox to Perform Computations on Graph," URL:http://www.ceremade.dauphine.fr/~peyre/, Toolbox_Graph.

[14] MeshLab: "An open source, portable, and extensible system for the processing and editing of unstructured 3D triangular meshes," URL: http://meshlab.sourceforge.net/ .