AN INTELLIGENT WATERMARKING ALGORITHM FOR 3D MODELS BASED ON FUZZY INFERENCE TECHNIQUE

[1] ZAINAB N. Al-QUDSY, [2] SHAIMAA H. SHAKER, [3] NAZHAT SAEED ABDULRAZQ

[1] Department of computer science,
Baghdad College of Economic Sciences University, Baghdad, Iraq
[2] Department of computer science, University of Technology, Baghdad, Iraq
[3] Departments of electro mechanical engineering,
University of Technology, Baghdad, Iraq

E-mail: [1]dr_zainabqudsi@baghdadcollege.edu.iq, [2]120011@uotechnology.edu.iq,
[3]drnazhatsaeed2016@gmail.com

ABSTRACT

In this paper, we propose new and robust 3D model watermarking method. Firstly, Fuzzy C-mean (FCM) algorithm clusters 3D mesh vertices into proper and improper regions for embedding the watermark to improve the imperceptibility. Afterward, the geometrical features are extracted using the mean curvature of vertices, area of triangular faces and angles between the faces normal and the average normal for rings around vertices. Further, the geometrical features are fed to fuzzy inference system (FIS) that give weights for vertices to indicate the relevance of the vertices to hiding the watermark and enhance the robustness. Finally, the watermark is blindly extracted without needing for the original models after the application of various types of attacks. The performance evaluation shows high imperceptibility and tolerance against attacks, such as addition of noise, smoothing, cropping, translation, and rotation. The proposed algorithm has been successfully designed, implemented, and tested.

Keywords: *3D model watermarking, Mean curvature, Geometrical properties, Fuzzy inference system, Fuzzy C-mean clustering.*

1. INTRODUCTION

Owing to the latest and rapidly growing graphic design technology, 3D models have emerged as new types of multimedia data. Generating 3D models requires special skills and great effort from designers. These models are quickly deployed in various areas, including Web applications, graphic design, video games, computer-aided design, military movies, and medical visualization [1]. In general, the current Internet provides users with high flexibility, which enables them to easily create perfect copies of digital multimedia and distribute them online without authorization from the actual owners. The use of watermarks is a solution that ensures authentication and copyright protection. Watermarking is a branch of information hiding used to hide secret information in multimedia data, such as digital images, text documents, audios, video clips, or 3D models. A digital watermark is a label embedded in multimedia data. This label acts as a digital signature, thereby indicating the data's ownership or authenticity. The optimal 3D model watermarking algorithm must meet two requirements: imperceptibility (i.e., the viewer cannot distinguish the original from the watermarked version) and robustness (i.e., the watermark cannot be possibly removed or altered and can still be recovered after different types of attacks). 3D watermarking can be applied either in spatial or frequency domains. In the frequency domain, the 3D models are represented in terms of their frequencies. In the spatial domain, the models are represented as a mesh of vertices connected by a set of edges to compose a set of faces, and each vertex can be represented by three Cartesian coordinates, namely, X, Y, and Z [2]. In general, working with the special-based technique with 3D is better than with the frequency-based one because many bits of watermark can be embedded into the original 3D model and it is robust to attacks. Furthermore, the frequency domain is more complex due to the irregular nature of the 3D model and requires long execution time due to the application of transform

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

and inverse transform [3]. The watermarking techniques can be classified as robust and fragile watermarks depending on their application. Robust watermarks are used for ownership and copyright protection so that they should resist against various types of attacks that try to remove or destroy the hidden watermark. On the contrary, fragile watermarks are used for authentication and detection of any illegal modification in the content so that they should be highly sensitive to any change. Watermarking algorithms are classified as blind and non-blind based on the existence of the original 3D model in the extraction process. Usually, the blind-type extraction is the inverse process of the embedding steps. In the non-blind type extraction, the hidden data are retrieved by measuring the difference or correlation between the original and watermarked 3D models.

From the invisibility aspect, one can assume that inserting the watermarks in less perceptible regions would reduced the chances of forgery by counterfeiters. But on the other hand, it is necessary to ensure that the watermark can be retrieved with the least amount of distortion after exposure to attacks.

The proposed method is depended on our previous works in [14,15],where different local geometrical properties of the vertices have been measured and used as a criteria for selecting the vertices for hiding process the watermark.in this work, Depending on the results that obtained from our previous works, a set of rules are proposed that used by Fuzzy inference system to assign the weight factor to all vertices. Depending on what has been stated previously, the main contributions of this work are: 1) We propose robust and blind 3D model watermarking method that capitalizes on the geometrical properties of the model regions and their mean curvature points, and benefiting from the robustness of flat points against attacks; 2) we introduce a new intelligent and an automatic watermark embedding and extraction approach by employing FIS for identifying the optimal domain for embedding the watermark to increase the robustness;3)We introduce evidence that intelligent watermark insertion implemented across the different regions enhance imperceptibility and reduce the distortion caused by hiding the watermark;4) We Employs the fuzzy C-mean to identify the optimal payload as increasing the size of the payload usually decreases the imperceptibility and robustness, a balance between these quantities needs to be set; 5) We present a comprehensive performance evaluation after applying many types of malicious and geometrical attacks, after that, we performed blind extraction watermark and comparing it with the state of the art methods.

2. RELATED WORKS

Many important studies in the field of watermark hides watermark in the 3D model. Hitendra and Agarwal [4] presented a non-blind watermarking algorithm based on dividing the vertices of a 3D mesh surface into flat, peak, and deep regions depending on the values of the mean curvature. Watermark bits are embedded in deep vertices that have less mean curvature values based on the hash code obtained by hashing the secure watermark to MD5 hash function. Their work is robust against various distortion attacks, such as smoothing, simplification and subdivision, and distortion-less attack. By contrast, Zhan *et al*. [5] proposed a blind watermarking technique based on the fluctuation values of the root-mean-square (RMS) curvature. The watermark is embedded by modulating the mean normalized fluctuation values of the RMS curvature. The experimental results showed that the algorithm has good imperceptibility and provided better resistance against vertex rearrangement, rotation, translation, uniform scaling, noise, smoothing, quantization, and simplification. Singh, *et al.* [6] introduced a non-blind watermark detection scheme. The proposed algorithm was based on the computation of the distance between each vertex and the center of gravity of the 3D object. The watermark was divided into patches with a size of 128 and hashed using MD5 with the sum of vertex normal in one group of vertices. Then, Exclusive or operation was performed by applying hash function and watermark bits. The result was embedded in the vertices of the mesh model. The proposed algorithm was resistant against translation, rotation, and uniform scaling attacks. Amar *et al.* [7] presented blind detection schema. The watermark bits were embedded by decreasing the Euclidian distance from the center mass of faces to the mass center of the mesh. The robustness of the proposed algorithm was improved by turbo encode to correct the errors caused by transmission or distribution. The performance evaluation showed good results in terms of imperceptibility and robustness. Researchers often employed artificial intelligence techniques to select appropriate places to embed watermarks in 3D

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

objects and obtain a 3D object watermarking algorithm that provides results with high imperceptibility and robustness. El Zein *et al.* [8] introduced a non-blind watermarking algorithm. The K-mean clustering technique was used to cluster 3D mesh vertices to find the best positions to embed the watermark. The clustering operation was performed on the basis of the computation of the angles between the normal and average normal surfaces that input as feature vector. The algorithm was robust against geometric attacks, such as cropping, adding noise, and smoothing, and also showed high results of imperceptibility. Laftah and Khalil [9] proposed a blind 3D mesh watermarking algorithm based on subdividing the faces of the mesh to remove noise and increase payload. The watermarking bits were encrypted by applied XOR with normal direction of every vertex. K-means clustering was applied to the vertices, and the cluster with the largest silhouette value was selected for the embedding process. The algorithm showed good imperceptibility and robustness against various types of attacks .Motwani *et al.* [10] presented a non-blind algorithm using a linear SVM to classify vertices as suitable or unsuitable candidates for watermark embedding. Feature vectors were derived from the curvature estimates of a one-ring neighborhood of vertices and used to train SVM. The robustness of the proposed algorithm showed acceptable resistance against smoothing, noise, and cropping attacks. Tamane and Deshmukh [11] represented a fragile, non-blind watermarking algorithm based on the exploited Haar wavelet transform (HWT) to convert the 3D model from special domain to frequency domain. The watermark bits were embedded in the low-resolution frequency coefficient after applying three levels of HWT. The watermark bits were scrambled by using Arnold transform, and the selected coefficient was hidden. The FIS was built to provide the weight factor to decide where the watermark will be inserted. The experimental results showed high imperceptibility and good robustness against cropping but poor robustness against smoothing and adding random or salt-and-pepper noise. The same researchers expanded their work in [12] by building a hybrid neuro- fuzzy system by using error backpropegation neural network to extract the features inputted to the FIS. Subsequently, the model was converted into the frequency domain by using HWT. The watermark data multiplied with the weight factor and was added into the coefficient of the middle-frequency sub-band. The performance evaluation showed high robustness at the expense of imperceptibility. The proposed schema was robust against adding random noise, cropping, scaling, and Gaussian low pass filter. In [13], the researchers proposed a non-blind watermarking algorithm based on utilizing C-means clustering. The vertex of the mesh clusters was either a suitable or unsuitable medium for embedding the watermark bits. The feature vector contains the angles between the normal and their average. The embedding was performed based on measuring the average and standard division for the vertices. The proposed method showed high robustness against cropping, smoothing, and adding noise attacks. In this study, the proposed algorithm is based on two algorithms proposed in our previous works [14, 15] with additional development. In [14] introduced a robust and blind watermarking method based on hiding the watermark in the vertices that have positive mean curvature values. While in [15] the proposed method was to study the effect of the area around the vertices of the 3D model. The study concluded that small area faces are better than the others. The method proposed in this paper was based on building the FIS to identify the optimal regions in the 3D model for hiding the watermark and also improving the robustness without affecting the visual quality of the original model.

3. PROPOSED METHOD

The general framework of 3D model watermarking system is constructed from the FIS and the embedding and extracting algorithms.

3.1 FIS

Fuzzy logic introduced by Lotfi A. Zadeh in 1965 that used to solve the problems without previous knowledge of its mathematical representation [17, 19]. The fuzzy logic provides the ability to handle the numerical data and linguistic knowledge. The idea of fuzzy logic employed by Ebrahim Mamdani in 1975 who built the first fuzzy control system that used to control the steam and solar engine. The Mamdani fuzzy inference system consists of the following four steps [11, 18]:

1. **Fuzzification** : is the process that convert the crisp inputs to fuzzy quantities by gives the degree of membership that represent

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

how much the crisp inputs belong to fuzzy sets.

2. **Rule evaluation:** is the process that applies the fuzzy inputs to the antecedents of the fuzzy rules that generated based on human experiences to obtain the consequents of the rules.

3. **Aggregation:** is the process that combined the fuzzy output from all rules to obtain a single fuzzy output.

4. **Defuzzification:** is the last process that converts the fuzzy output from aggregation the output of the rules to single crisp output.

In this paper, Mamdani fuzzy inference system is built to assign the weight factor (WF) for all the vertices of the 3D model based on a set of rules. The WF is used to decide whether the vertices are suitable or unsuitable candidates for the embedding process The fuzzy inputs to the FIS are a set of geometrical properties of the vertices that are observed and calculated: mean curvature (MC), area of the ring around the vertices, and sum of angles in the ring that have six faces. The MC exploits the proposed watermarking method in [14]. In this work, the method proved and concluded that hiding watermark bits in vertices with positive mean curvature values was less sensitive to the human visual system and provided high imperceptibility and robust performance results. In [15], we investigated the effect of the area of faces around the vertices in embedding the watermark. The study proved that the small area rings are suitable domain for embedding. In this study, the sum of the angles around the vertices in one ring of valance six is calculated to be fed to the FIS with the MC and ring area. Usually, the 3D model is composed of a set of rings. The ring valance is the number of faces around the central vertex. Figure1 shows the angles (θ)between the faces normal (n_i) and average normal ($navg$) around the central vertex in one ring of valance six. The mushroom model and its faces normal are shown in Figure 2.

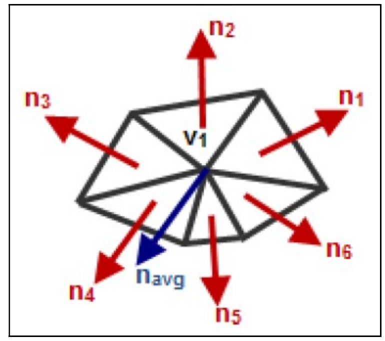

Figure 1: The one ring of valance six

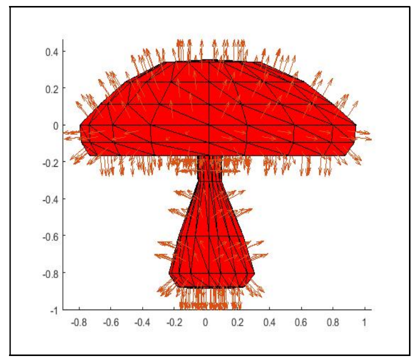

Figure 2: The faces normal

Each ring has six angles, which are calculated based on following equations:

$$\theta_i = \frac{1}{\cos\left(\frac{n_i.navg}{|n_i|.|navg|}\right)} \quad (1)$$

$$navg = \sum_{i=1}^{6} \frac{n_i}{6} \quad (2)$$

The MC, ring area, and sum of angle values are fed as fuzzy inputs to FIS, and the weight factor for each vertex is determined by FIS as output. The proposed FIS is shown in Figure 3.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

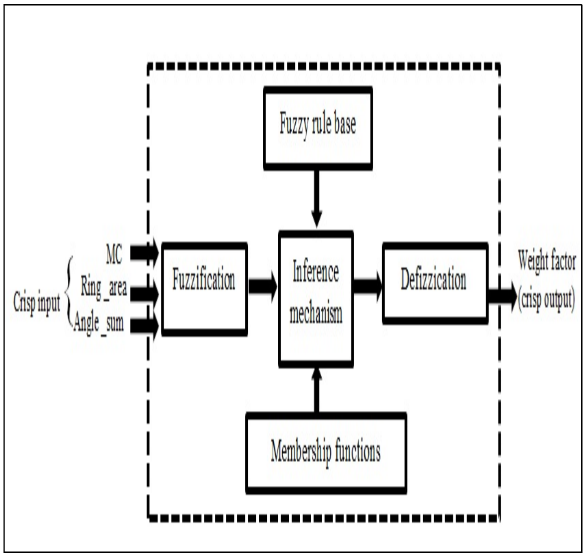

Figure 3: FIS

3.1.1 Membership functions

1. Mean curvature for vertices

Each vertex has an MC value that may be positive, zero, or negative. The experiment shows that the vertex with a positive mean curvature value is the best candidate for hiding the watermark [14]. The vertex can be classified as flat, peak, and deep depending on its MC value. Figure (4) shows the fuzzy membership values for MC input.

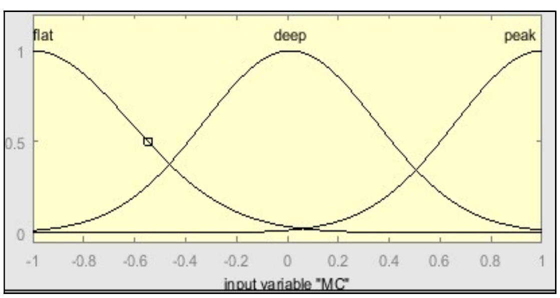

Figure 4: Membership function plots for MC

2. Ring area around vertex (ring _area)

Each vertex is surrounded by a number of faces that form one ring around the vertex. The ring area is calculated by taking the sum of the areas for all faces around the vertex. The values of the ring area are classified into three categories (low, moderate, and high). Figure (5) illustrates the smooth, moderate, and rough area values as fuzzy input.

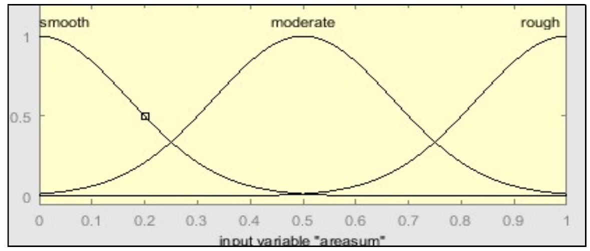

Figure 5: Membership function plots for ring _area

3. Angles around vertex (angle_ sum)

The angles that surround the vertex in one ring of degree six are classified as small, medium, and large. Only the medium angles are selected as suitable domains for hiding the watermark. Figure (6) shows the plots of fuzzy input for angle values.

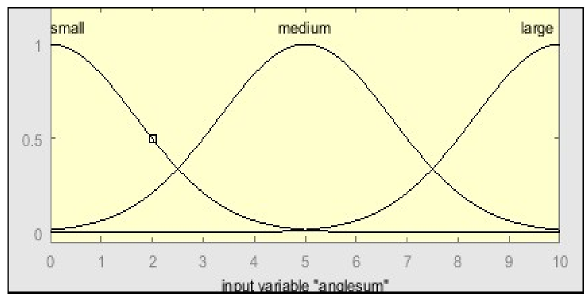

Figure 6: Membership function plots for angle_ sum

4. Weight factor (WF)

The fuzzy output from FIS is the WF that can take five values (least, less, average, high, and highest), which are plotted in Figure (7). Only the vertices with high or highest WF values are accepted and selected to hide watermarks while the others are discarded to improve imperceptibility and robustness.

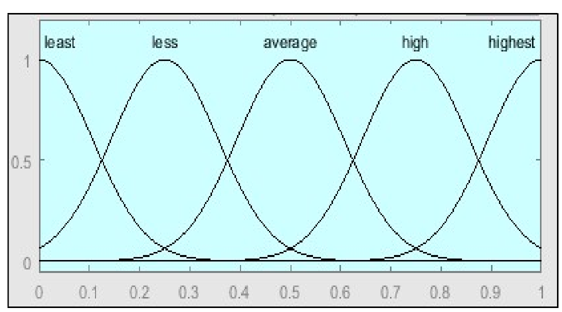

Figure 7: Membership function plots for WF

3.1.2 Fuzzy rule base

A set of fuzzy rules are generated and stored as knowledge based on the inference engine. The rules are created on the basis of three facts:

1) The vertex that belongs to the flat region is less sensitive to attacks.
2) The vertex that has small area faces around it in one ring of valance six is a good candidate for embedding.
3) The HVS is less sensitive toward the vertex with the sum of angles around it.

These rules are listed as follows:

Rule 1: if (MC is flat) and (ring _area is smooth) and (angle_ sum is small) then (wf is highest)

Rule 2: if (MC is flat) and (ring _area is moderate) and (angle_ sum is small) then (wf is high)

Rule 3: if (MC is flat) and (ring area is rough) and (angle_ sum is small) then (wf is high)

Rule 4: if (MC is deep) and (ring _area is smooth) and (angle_ sum is small) then (wf is high)

Rule 5: if (MC is deep) and (ring _area is moderate) and (angle_ sum is small) then (wf is average)

Rule 6: if (MC is deep) and (ring _area is rough) and (angle_ sum is small) then (wf is average)

Rule 7: if (MC is peak) and (ring _area is smooth) and (angle_ sum is small) then (wf is high)

Rule 8: if (MC is peak) and (ring _area is moderate) and (angle_ sum is small) then (wf is average)

Rule 9: if (MC is peak) and (ring _area is rough) and (angle_ sum is small) then (wf is average)

Rule 10: if (MC is flat) and (ring _area is smooth) and (angle_ sum is medium) then (wf is high)

Rule 11: if (MC is flat) and (ring _area is moderate) and (angle_ sum is medium) then (wf is average)

Rule 12: if (MC is flat) and (ring _area is rough) and (angle_ sum is medium) then (wf is average)

Rule 13: if (MC is deep) and (ring _area is smooth) and (angle_ sum is medium) then (wf is average)

Rule 14: if (MC is deep) and (ring _area is moderate) and (angle_ sum is medium) then (wf is less)

Rule 15: if (MC is deep) and (ring _area is rough) and (angle_ sum is medium) then (wf is less)

Rule 16: if (MC is peak) and (ring _area is smooth) and (angle_ sum is medium) then (wf is average)

Rule 17: if (MC is peak) and (ring _area is moderate) and (angle_ sum is medium) then (wf is least)

Rule 18: if (MC is peak) and (ring _area is rough) and (angle_ sum is medium) then (wf is average)

Rule 19: if (MC is flat) and (ring _area is smooth) and (angle_ sum is large) then (wf is high)

Rule 20: if (MC is flat) and (ring _area is moderate) and (angle_ sum is large) then (wf is average)

Rule 21: if (MC is flat) and (ring _area is rough) and (angle_ sum is large) then (wf is average)

Rule 22: if (MC is deep) and (ring _area is smooth) and (angle_ sum is large) then (wf is high)

Rule 23: if (MC is deep) and (ring _area is moderate) and (angle_ sum is large) then (wf is less)

Rule 24: if (MC is deep) and (ring _area is rough) and (angle_ sum is large) then (wf is less)

Rule 25: if (MC is peak) and (ring _area is smooth) and (angle_ sum is large) then (wf is average)

Rule 26: if (MC is peak) and (ring _area is moderate) and (angle_ sum is large) then (wf is least)

Rule 27: if (MC is peak) and (ring _area is rough) and (angle_ sum is large) then (wf is least)

3.2 Watermark Embedding Algorithm

As preprocessing, the vertices of the 3D model are clustered to two clusters using fuzzy C-mean clustering. Fuzzy c-mean clustering is the fuzzy version of the k-mean clustering in which each vertex belong to more than one cluster and assigned with degree of membership that show how much the vertex belong to the cluster. The block diagram for the proposed embedding algorithm is illustrated in figure 8.The clusters validated to measure their

Journal of Theoretical and Applied Information Technology
15th May 2019. Vol.97. No 9

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

goodness and the cluster cohesion (WSS) is used as validation measurement to identify how the vertices in specific cluster are closely related to each other. The cohesion is computed based on the following formula:

$$WSS = \sum_{i=1}^{k} \sum_{v \in C_i} (v - m_i)^2 \quad (3)$$

where K represents the number of vertices in cluster c_i and m_i is the centroid of cluster c_i. The steps of the proposed algorithm for hiding the watermark in the original 3D model are explained as follows:

Input: 3D model stored in .off file format, secret text as a watermark

Output: 3D watermarked model

Step 1: Read 3D model from .off file format.

Step 2: Convert 3D model to triangle mesh.

Step 3: Read vertices and faces from 3D model and store them in faces (F) and vertices (V) Matrices, respectively.

Step 4: Read secret text and reshape it to binary vector W= [w1...wn], where n is the length of the secret text.

Step 5: Compute mean curvature (MC) for all vertices. The value of mean curvature is positive, zero, or negative.

Step 6: Compute the valance of each ring around each vertex.

Step 7: Compute the sum of triangle area (sum_area) for the rings that have six faces.

Step 8: Compute the sum of angles (sum_angle) around the vertex that has valance six.

Step 9: Input the geometrical properties (MC, sum_angle, sum_area) to FIS to obtain the weight factor (WF) for all vertices.

Step 10: Cluster 3D model vertices by applying C-mean clustering**.**

Step 11: Evaluate the clusters by using equation number 3 and select the best cluster that has minimum cohesion value.

Step 12: Select the vertex from the selected cluster that has WF greater than or equal to 0.7 for embedding.

Step 13: Embed one bit from W[i] by modifying the z coordinate for the selected vertex depending on the following equations:

$$V(new) = V(old) + W[i] \quad (4)$$

where $V(new)$ represents the watermarked vertex, $V(old)$ represents the original selected vertex.

Step 14: Repeat steps 12 and = 13 until all secret data are embedded.

Step 15: Save and display watermarked model.

3.3 Watermark Extraction Algorithm

The watermark extraction algorithm is inversed to the embedding process. It is a blind type. Thus, the watermark is extracted without the need for the original 3D model in the extraction process. The proposed extraction algorithm is described in detail in the following steps:

Input: Watermarked model

Output: Watermark (secret text)

Step 1: Read 3D watermarked model from .OFF' File format (M).

Step 2: Convert 3D watermarked model to triangular mesh.

Step 3: Read vertices and faces from the mesh and store them in (V) and (F) metrics.

Step 4: Select the vertices that have six faces and compute its MC, sum_area, sum_angles.

Step 5: Input MC, sum_area, and sum_angles to FIS to obtain the weight factor.

Step 6: Select the vertex that has weight factor greater than or equal to 0.7

Step 7: I=1

Step 8: Extract one bit from the selected vertex and store them in the watermark vector.

Step 9: I=I+1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Step 10: Repeat steps 8 and 9 until I > length (secret text)
Step 11: Convert watermark vector from binary to text.

Step 12: Display watermark.

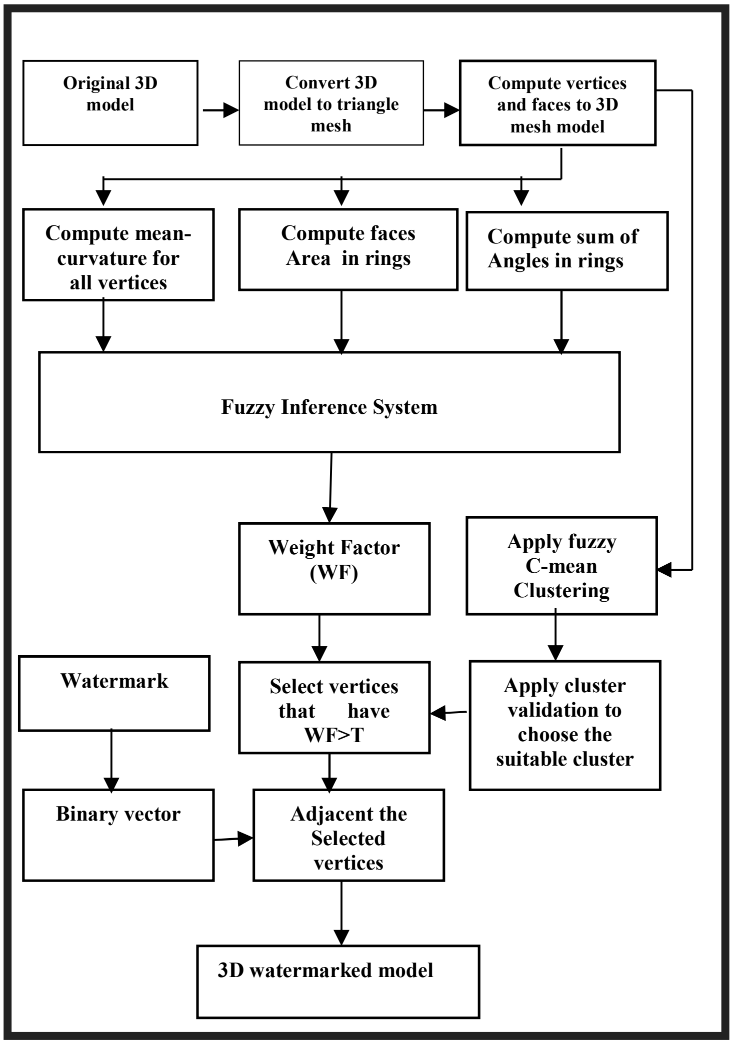

Figure 8: Block Diagram for embedding algorithm

4. EXPERIMENTAL RESULTS AND DISCUSSION

The performance efficiency is usually verified according to imperceptibility, robustness, and payload metrics. Five 3D models with different natures of surface and properties are used as the testing set. The text of size 151 bits is used as watermark. The next sections summarize the experimental results.

4.1 Imperceptibility and payload measurements

Imperceptibility shows the degree of the difference between the original and watermarked 3D mode. vertex signal-to-noise ratio (VSNR) is used to measure imperceptibility and it should be as high as possible [16]. The imperceptibility results show high measurements to all the tested models, as explained in Table I, with the storage details of each one. VSNR is calculated based on the following equations[17]:

$$SNR=\frac{\sum_{i=1}^{N}(x_i^2+y_i^2+z_i^2)}{\sum_{i=1}^{N}[(x_i-x_i')^2+(y_i-y_i')^2+(z_i-z_i')^2]} \quad (5)$$

where N is the number of vertices; x_i, y_i, and z_i are the coordinates of the vertices in the original model; and x_i', y_i', and z_i' are the new watermarked coordinates.

$$VSNR=20\log10(SNR) \quad (6)$$

Payload refers to the maximum size of bits that can be used as the watermark. In Table (1), the experimental results show high imperceptibility measurements to all tested models. The payload1 and payload2 are measured when the WF equal to 0.7 and 0.6 respectively. The experimental results show that the payload increases when the weight factor decreases.

Table 1: Imperceptibility measurements

Model name	Faces	Vertices	Payload2	Payload1	VSNR
Mushroom	448	226	45	155	146.20
Venus	1396	711	367	659	153.00
Cow	1448	726	569	651	129.89
Camel	1168	586	425	498	142.26
Nefertiti	562	299	72	167	149.13

4.2 Robustness measurements

A robust watermark should survive against a wide variety of attacks. The correlation factor (CF) is used to verify the robustness against those attacks and it is calculated based on the following equation:

$$CF = \frac{\sum_{i=1}^{N}(W_i-\overline{W_i})(E_i-\overline{E_i})}{\sqrt{\sum_{i=1}^{N}(W_i-\overline{W_i})^2 \sum_{i=1}^{N}(E_i-\overline{W_i})^2}} \quad (7)$$

Where $\overline{W_i}$ and $\overline{E_i}$ indicate the average of the inserted watermark bit sequence and the average of the extracted watermark bit sequence, respectively.

The CF value equal to 0.7 or more is a good degree of survival for the hidden watermark. Tables 2 and 3 show the results of CF, which measures the similarity between the original and extracted bits from attacked watermarked models. Five geometrical and removable attacks are simulated and applied to the 3D watermarked model as follows:

1) **Rotation** is the process of rotating a 3D object around one coordinate X, Y, or Z to a specific angle in different directions. Figure 9 shows Venus model after rotating at 240° around the Z-coordinate.

2) **Scaling** is the process of maximizing or minimizing the size of a 3D model by a fixed scale factor in all directions. Figure 9 shows the mushroom model after applying scaling where the scaling factor is equal to-0.5.

3) **Translation** is the process of moving the 3D model toward a specific direction. Translation adds a constant vector P to every vertex in the model. Figure 10 shows cow model after applying translation, where P= -0.5.

4) **Subdividing** is the process of creating additional faces by dividing each face into N*N small faces, where N is an integer number >1. Figure 11 shows the camel model after applying subdivision, where N = 5.

5) **Adding Noise**: All vertices in the 3D watermarked model are affected after the same level of noise was added. Furthermore, increase in noise reduces the imperceptibility of the hidden watermark. Figure 12 shows the Nefertiti model after adding noise level=0.0001

6) **Smoothing**: is the process that replaces the vertices of the 3D model by the mean of their neighbors. Figure 13 shows the cows model after applying smoothing, where N = 1.

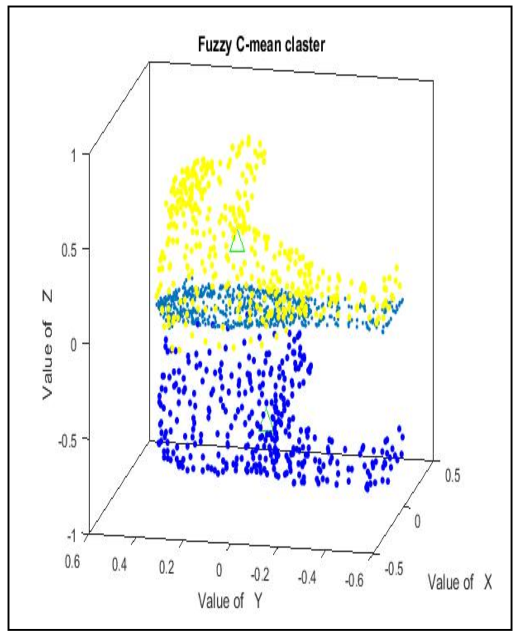

Figure 8: Fuzzy C-means Clustering

Table 2. Correlation results after applying geometrical attacks

Model name	Rotation			Scaling			Translation		
	70	120	240	2	3	4	−5	2	5
Mushroom	1	1	1	1	1	1	1	1	1
Venus	1	1	1	1	1	1	1	1	1
Cow	1	1	1	1	1	1	1	1	1
Camel	1	1	1	1	1	1	1	1	1
Nefertiti	1	1	1	1	1	1	1	1	1

Table 3. Correlation results after applying malicious attacks

Model name	Subdivide			Smoothing			Adding Noise		
	2	3	4	1	0.9	0.8	0.001	0.0001	0.00001
Mushroom	1	1	1	0.0373	1	1	0.103	0.6266	0.93
Venus	1	1	1	-0.0373	1	1	0.233	0.8432	0.96
Cow	1	1	1	0.1879	1	1	0.753	0.9389	1
Camel	1	1	1	-0.1914	1	1	0.374	0.7548	0.93
Nefertiti	1	1	1	0.0039	1	1	0.007	0.7851	0.96

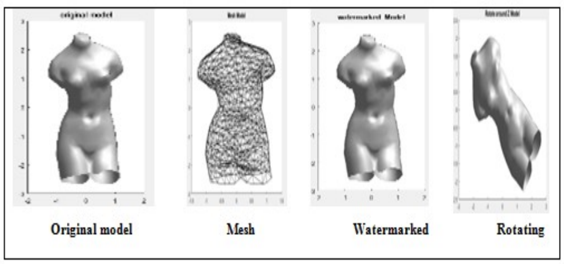

Figure 9: watermarked Venus Model after rotating

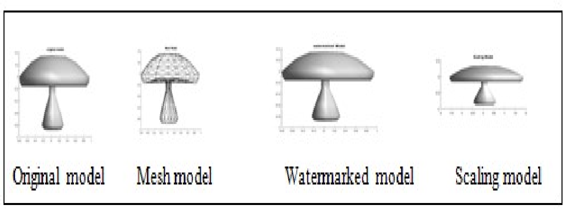

Figure 10: Watermarked Mushroom Model after Scaling

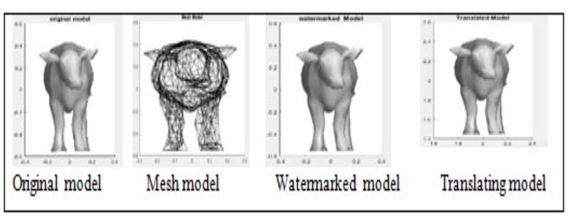

Figure 10: Watermarked Cow Model after translating

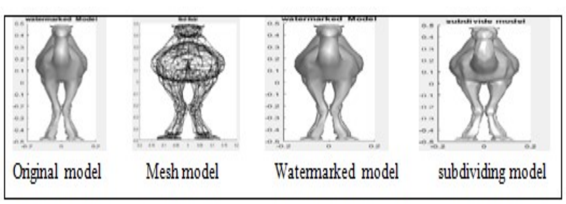

Figure 11: Watermarked Camel Model after subdividing

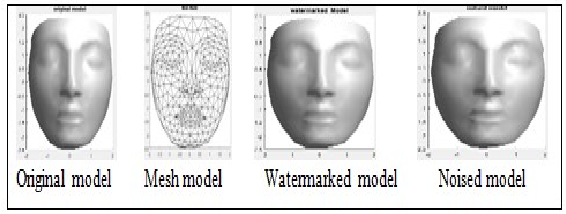

Figure 12: Watermarked Nefertiti Model after adding noise

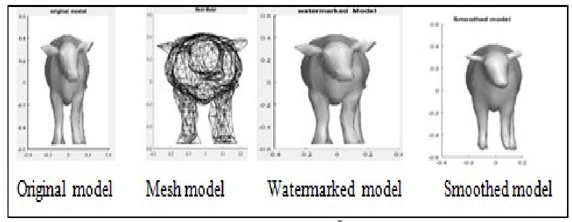

Figure 13: Watermarked Cow Model after Smoothing

The experimental results provide high imperceptibility and robustness by employing the FIS as an artificial intelligent technique to automatically identify the suitable vertices for embedding algorithm. The hidden watermark is fully recovered before and after various types of attack, such as rotating, scaling, translation, and subdividing, adding noise and smoothing at specific levels.

Table 4.Results of comparison the proposed approach with related works

Referenc	methodology	VSNR	Robustness against
[14]	Blind based on mean curvature	138	rotation, cropping, translation, noise, and smoothing
[15]	Blind based on geometrical properties	151	rotation, cropping, translation, and noise addition
[12]	Non-blind Based on (FCM)	140	Similarity transforms, noise addition,
[13]	Non-blind based on FIS	133	rotation, scaling, and smoothing
Our approach	Blind based on (FIS)	153	rotating, scaling, translation ,smoothing, noise and subdividing

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

According to the results of comparison our proposed approach with o other related works as shown in Table 4, we conclude that the proposed approach has achieved good results in terms of imperceptibility and robustness.

5. CONCLUSIONS

In this study, a new robust and blind 3D model watermarking method was proposed for copyright protection purposes. The vertices of 3D mesh have been pre-processed by applying FCM clustering algorithm to cluster them to suitable and unsuitable regions for embedding process. A set of geometrical properties, such as the mean curvature of vertices, area of triangular faces and angles between the faces normal and the average normal for rings around vertices were computed and fed to the FIS to assign the weight factor for each vertex. According to these weights, the proposed embedding algorithm automatically selected the relevance vertices and the volume of payload that ensure high imperceptibility and robustness results .further, The watermark has been carefully hidden in the vertices of the 3D model based on the intelligent decision from the FIS. Moreover, the experimental results proved that the proposed algorithm had a high level of robustness against different kinds of attacks, such as rotation, scaling, translation, noise, smoothing and subdividing. The robustness results obtained for attacked watermarked models indicate that our method is an intelligent approach that capitalizes on the invariance property of geometrical features for vertices.

REFRENCES:

[1] G. Fabio, C. Massimiliano, P. Sumanta, and, B. Marco Di "Introduction to computer graphics," *Reading: Addison-Wesley*, Vol.55, 1994.

[2] K. McHenry and B. Peter, "An overview of 3d data content, file formats and viewers." *National Center for Supercomputing Applications*, 2008, 1205: 22.

[3] K. Wang, G. Lavoué, F. Denis and A. Baskurt,."A comprehensive survey on three-dimensional mesh watermarking." *IEEE Transactions on Multimedia*, Vol. 10, 2008,pp. 1513-1527

[4] G.Hitendra, and S. Agarwal."A secure image based watermarking for 3d polygon mesh", *Science and Technology*, Vol. 16, 2013,pp. 287-303.

[5] Y. Zhan, Y. Li, X. Wang, Y. Qian, "A blind watermarking algorithm for 3D mesh models based on vertex curvature," *Journal of Zhejiang University SCIENCE,* Vol. 15, 2014,pp. 351-362

[6] L. K. Singh, D. Chaudhry and G. Varshney, "A novel approach of 3d object watermarking algorithm using vertex normal." *International Journal of Computer Applications*, Vol. 60, 2012.

[7] Y. Amar, I. Trabelsi, N. Dey and M. Bouhlel, "Euclidean Distance Distortion Based Robust and Blind Mesh Watermarking*, International Journal of Interactive Multimedia and Artificial Intelligence*,Vol. 4, 2016,pp. 46-51.

[8] O.M. El Zein, L. El Bakrawy,, and N. Ghali, "A NON-BLIND ROBUST WATERMARKING APPROACH FOR 3D MESH MODELS", *Journal of Theoretical and Applied Information Technology*,Vol. 83, 2016.

[9] M. Laftah and M. Khalil, "Data Hiding in 3D Model Using Artificial Techniques." *International Journal for Sciences and Technology*,Vol. 143, 2016,pp.1-9.

[10] R. Motwani, M. Motwani, B. Bryant, F. Harris Jr and , A. Agarwal,"Watermark embedder optimization for 3D mesh objects using classification based approach", *Signal Acquisition and Processing, 2010. ICSAP'10. International Conference on*. IEEE, 2010.

[11] S. C. Tamane and R. R. Deshmukh, "Blind 3D Model Watermarking Based On Multi-Resolution Representation and Fuzzy Logic", *International Journal of Computer Science and Information Technology (IJCSIT)*, Vol. 4, 2012, pp. 117-126.

[12] S. C. Tamane and R. R. Deshmukh, "Watermarking 3d surface models into 3d surface models based on ANFIS"*Advances in Computing*, Vol.*2*, 2012, pp.29-34.

[13] O. M. El Zein, L. M. El Bakrawy and N. I. Ghali,"A Robust 3D Mesh Watermarking Algorithm utilizing Fuzzy C-Means" Clustering, *Future Computing and Informatics Journal,*Vol. 2,2017,pp. 148-156.

[14] Z. N. Al-Qudsy, S. H. Shaker and N. S. Abdulrazzque," Robust Blind Digital 3D Model Watermarking Algorithm Using Mean Curvature", In *International Conference on New Trends in Information and Communications Technology Applications*, Springer, Cham., 2018,pp. 110-125.

ISSN: **1992-8645** www.jatit.org E-ISSN: **1817-3195**

[15] Z. N. Al-Qudsy, S. H. Shaker and N. S.," Proposed an efficient blind digital 3D modelwatermarking algorithm using geometrical properties. *Interciencia Journal* .Vol 43, 2018, pp. 110-124.

[16] J. Dugelay, A. Baskurt and M. Daoudi, 3D *object processing: compression, indexing and watermarking*. John Wiley and Sons, 2008.

[17] E. Volna, Introduction to soft computing, 2013.

[18] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, Vol. 7, No. 1,1975, pp. 1-13.

[19] L. A. Zadeh,, "Fuzzy sets," *Information and control*, vol. 8, no. 3, 1965, pp. 338-353.