Blind, Secured and Robust Watermarking for 3-D Polygon Mesh using Vertex Curvature

Priyanka Singh, K Jyothsna Devi
SRM University - A.P
Amaravati 522 502
A.P, India

Abstract—In this paper a blind, imperceptible, robust and secure watermarking scheme for 3-D mesh models is presented. Here, the watermark is embedded in deeper surface vertices to minimize the perceivable distortion. Deeper surface vertices are selected on the basis of their mean curvature (lesser than zero) and converted to spherical coordinates. Out of the three spherical coordinates, radial distance represents approximate mesh and is invariant to distortionless attack. Therefore, watermark bits are embedded by modifying the distribution of radial distance to make the proposed scheme robust against such attacks. Radial distances are divided into bins and normalized to range in [0, 1]. Each bin accommodates one watermark bit. Watermark is embedded repeatedly in the 3-D mesh to resist cropping and simplification attack. To ensure higher security, a 128-bit unique watermark is generated by hashing (MD5 algorithm) the mean of histogram map obtained from a grayscale watermark image. Watermark bits are extracted from bins by comparing mean of each bin with a reference value. Since original mesh is not required at the time of extraction, the proposed scheme is blind. Through experimental results, it is demonstrate that the proposed scheme has good visual masking and higher robustness against various attacks. It shows improved performance as compared to some of the prominent schemes.

Keywords—3-D mesh watermarking; mean curvature; radial distance; spherical coordinates; visual masking

I. INTRODUCTION

In the past few decades there has been extensive growth in the use of 3-D objects in virtual reality, video games, animation, medical sciences, industries, and computer aided design (CAD), architecture, archaeology and scientific data visualization [1]. With the increasing accessibility and wide distribution of 3-D objects, the concern for its copyright protection has also increased. Digital watermarking is considered as a potential solution for copyright protection of various multimedia contents [2]. Therefore, researchers are exploring 3-D watermarking techniques for the copyright protection of 3-D mesh objects. Lot of research has been done in audio, image and video watermarking in spatial as well as transform domain. But, the watermarking schemes developed for other multimedia contents cannot be readily used for 3-D mesh due to lack of unique representation. 3-D objects are generally represented by polygonal meshes. Fig. 1 shows 3-D bunny model represented by triangular polygon mesh. 3-D mesh has three basic elements: vertices, edges, and faces [3]. Vertex is the co-ordinate in 3-D space. Edges are connectivity between two vertices. Face describes connectivity of vertices to form smallest unit of 3-D mesh. There is no well defined order and connectivity of vertices, which makes watermark extraction very difficult. Moreover, number of vertices where watermark bits can be inserted are limited as compared to other multimedia content. Due these challenges, 3-D mesh watermarking is still in its infancy [4].

There are different classifications for 3-D mesh watermarking techniques [5]. According to the insertion space it is classified into spatial and transform domain. In spatial domain, the watermark is inserted directly by altering the coordinates of the mesh vertices. But, in the transform domain watermark is inserted by altering the transform coefficients. Spatial techniques provide higher imperceptibility but are less robust. Some robust spatial watermarking schemes have been reported in [6]– [8]. But, the main drawback of these schemes is that they are relatively less robust to connectivity attacks. Whereas, transform domain techniques are robust but has higher computational time. Watermarking techniques can also be classified into non-blind and blind depending on whether the original host is required or not at the time of extraction. Non-blind schemes require original mesh during watermark extraction in contrast to blind schemes. Generally, non-blind techniques are more robust than blind techniques. Whereas blind schemes have high capacity, and can be easily implemented. But, practical applications prevent access to the original 3-D model during extraction. Considering pros and cons of different types of watermarking schemes, robust blind watermarking schemes in spatial domain are more feasible for practical applications. The majority of the existing 3D watermarking schemes suffers with low imperceptibility, robustness, security and high computational cost. For real time applications the computational cost is assumed to be low. Hence there is need to develop more blind and robust
schemes for 3-D watermarking. In this paper a blind and robust 3-D watermarking scheme is presented which is robust to majority of attacks and maintains high imperceptibility. The rest of paper has following organization: Related work is reviewed in Section 2. The proposed 3-D watermarking scheme is described in detail in Section 3. Section 4 presents the experimental result and discussion. Section 5 concludes the paper.

II. RELATED WORK

Various 3-D mesh watermarking schemes have been proposed till date but still this field is in its infancy as compared to the development in image, audio and video watermarking. The two main hurdles in handling 3-D mesh watermarking are: lack of an ordered structure precludes systematic analysis of model and difficulty in anticipating all the attacks due to various mesh manipulation graphic tools. In spite of these challenges, researchers have put persistent efforts to propose various blind and non-blind 3-D mesh watermarking approaches for improving imperceptibility, robustness and payload.

In this direction, a blind scheme is proposed by Nakazawa et al. [9]. It uses mesh saliency based on the surface curvature to segment the triangular 3-D mesh. Then watermark bits are inserted in the regions by modulating the vertex norms. A similar approach is suggested by Zhan et al. [10] where watermark bits are interleaved in vertex bins formed on the basis of vertex curvature. Authors assert that this method has higher robustness and imperceptibility than the schemes proposed in [11], [12]. Li et al. [13] suggested to form bins of eigen values (distance between vertices to the model centre) based on improved vertex grouping. It uses piecewise mapping function to insert watermark bits in bins. It shows superior performance than [11]. Another blind scheme is proposed by Bata et al. [14] based on sparse QIM and run- length modulated LDPC. This scheme provides higher imperceptibility and lower BER in extracted watermark. Alaa et al. [15] proposed a blind scheme based on vertex distribution and skefinedness measure. Skewness measure is altered to embed the watermark. Authors claim that the scheme has good visual masking and resistant to a variety of common attacks. Recently, Sharma and Panda [7] have proposed a blind scheme and attempted to fulfill all the watermarking requirements by exploring Local curvature estimation and statistical characteristics for hiding watermark. Medimegh et al. [16] proposed a statistical 3-D watermarking scheme explored invariance property of salient points. They embedded statistical signature at different regions self-segmented around salient points. Hamid et al. [17] have proposed to a blind watermarking method in transform domain, where watermark is embedded in quantified wavelet coefficients using QIM based on 3-D mesh saliency. Though the scheme achieves good imperceptibility and robustness but, it is not robust against cropping attack. Other blind schemes has been reported in [18]– [21]. Some of the non-blind watermarking schemes are proposed in [22]– [25]. Although non-blind schemes exhibit satisfactory robustness but, mostly practical applications prevents having access to the original 3-D model during extraction. Thus, blind schemes are practically more feasible. From the literature review it is observed that the existing schemes are robust to either one or the other set of attacks. Majority of existing schemes have focused on achieving higher imperceptibility and robustness but have overlooked an important watermarking characteristics i.e. security. A comprehensive comparison of the related work is shown in Table I.

Motivation and contribution of the proposed work

From the literature survey, we have observed that there is prospect to develop blind 3D watermarking scheme which is robust to majority of attacks, has higher imperceptibility and security. For practical application of the scheme it should be computationally inexpensive. Motivated by these issues we propose a 3-D watermarking scheme having following contribution:

- **Improved robustness:** the proposed blind scheme withstands majority of common attacks like smoothing, additive noise, similarity transformation, quantization, cropping and simplification as evident from experimental results. It successfully survives Meshbenchmark attacks. The proposed scheme has higher robustness as compared to the existing popular schemes in [10], [13], [17]. Although robustness of the proposed scheme against noise attack is not at par with [10], [13], [17] scheme but has acceptable correlation values as discussed in section IV.

- **Improved visual quality:** watermark embedding in deeper vertices of 3-D mesh surface induces less perceivable distortion than the vertices of flat or peak surface [26]. Making use of this observation, in the proposed scheme we select deeper surface vertices for embedding watermark bits to minimize induced distortion. Deeper surface vertices are classified on the basis of mean curvature ($C_m$) [26]. Proposed scheme offers superior visual masking effect than schemes proposed in [10], [13], [17].

- **Enhanced security:** To enhance the watermark security, a 128-bit binary chaotic watermark sequence is embedded in the 3-D model. This sequence is generated from a grayscale watermark image/logo. Mean of histogram map of the grayscale watermark image is calculated. Then the mean is hashed by using MD5 hash algorithm to obtain the watermark sequence. Watermark sequence is highly sensitive to the change in watermark image pixels, thus ensures high watermark security.

- **Low computational cost:** The proposed scheme embeds in spatial domain therefore it is less computationally expensive. The proposed embedding scheme has linear time complexity as discussed further in section IV.C.

III. PROPOSED WATERMARKING SCHEME

The proposed 3-D mesh watermarking scheme is based on modulation of radial distance distribution. Proposed scheme consists of three modules i.e. watermark generation, embedding and extraction, explained in detail in the following subsections.
TABLE I. COMPARATIVE STUDY OF 3-D WATERMARKING SCHEMES(SM-SMOOTHING, SP-SIMPLIFICATION, SD-SUBDIVISION, CP-CROPPING)

<table>
<thead>
<tr>
<th>Method</th>
<th>Domain</th>
<th>Robustness against attacks</th>
<th>Geometric Connectivity</th>
<th>Blind</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jing et al. [24]</td>
<td>Spatial</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>64 bits</td>
</tr>
<tr>
<td>Song Li et al. [13]</td>
<td>Spatial</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>≤ 3%</td>
</tr>
<tr>
<td>Cho et al. [11]</td>
<td>Spatial</td>
<td>No</td>
<td>&lt;90%</td>
<td>No</td>
<td>64 bits</td>
</tr>
<tr>
<td>Garg and Agrawal</td>
<td>Spatial</td>
<td>No</td>
<td>&lt;70%</td>
<td>No</td>
<td>256 bits</td>
</tr>
<tr>
<td>Seung et. al. [19]</td>
<td>Spatial</td>
<td>Yes</td>
<td>≤30%</td>
<td>No</td>
<td>16 bits</td>
</tr>
<tr>
<td>Alaa et. al. [15]</td>
<td>Spatial</td>
<td>No</td>
<td>≤70%</td>
<td>No</td>
<td>32 bits</td>
</tr>
<tr>
<td>Zhan et al. [10]</td>
<td>Spatial</td>
<td>Yes</td>
<td>&lt;90%</td>
<td>No</td>
<td>16 bits</td>
</tr>
<tr>
<td>Hamidi [17]</td>
<td>Spectral</td>
<td>Yes</td>
<td>≤70%</td>
<td>No</td>
<td>64 bits</td>
</tr>
<tr>
<td>Feng et. al. [18]</td>
<td>Spectral</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>36 bits</td>
</tr>
<tr>
<td>Proposed</td>
<td>Spatial</td>
<td>Yes</td>
<td>≤50%</td>
<td>No</td>
<td>128 bits</td>
</tr>
</tbody>
</table>

A. Watermark Generation

In the proposed scheme, to ensure higher security a unique 128-bit binary chaotic watermark sequence is generated according to the grayscale watermark/logo image of size $M \times M$. Firstly, a histogram map (HM) of grayscale watermark is generated. Then the mean of HM is calculated which is further used to generate an intermediate key. The resulting key is hashed by the hash function MD5 to obtain a 128-bit random binary sequence. This sequence is embedded in the 3-D model. Other hash algorithms like SHA can also be used for hashing, but its computation time and security should be taken into account. Watermark sequence is highly secure as the secret key is highly sensitive to the minor alteration in the watermark pixels. The watermark sequence can resist brute force attack. During extraction, the authorised user can obtain the secret key according to the valid watermark image/logo provided for validation. Following are the steps for watermark generation:

**Step 1:** Generate the histogram map from the grayscale watermark image using transform function as follows:

$$HM = \sum_{e=1}^{M} (Histpw_e||Pw_e), \forall Pw[0, 255]$$

where Histpw is the frequency of a pixel pw in watermark image.

**Step 2:** Generate intermediate key (8-bits of binary key sequence) from the HM as:

$$Transformfunc = \sum_{e=1}^{M} dec(HM_e) \mod 2 \times 2^{14}$$

$$Intermediatekey = mean(Transformfunc)$$

where dec is decimal equivalent of HM.

**Step 3:** The resulting Intermediatekey (8-bits) is hashed by MD5 hash function. A 128-bits binary sequence (W) is generated by hashing. W is used as a binary watermark sequence for embedding.

B. Watermark Embedding

Proposed watermark embedding scheme is based on the modulation of radial distance ($\rho$) distribution in the deeper surface of mesh to reduce perceivable distortion and achieve higher robustness. Deeper surface vertices are selected on the basis of $C_m$. Vertices with $C_m < 0$ are deeper vertices. Fig. 2 shows grayscale representation of $C_m$ for bunny model, the darker area represents deeper vertices ($C_m < 0$) used for inserting the watermark, and the lighter area shows superficial vertices ($C_m > 0$).

Selected vertices are converted from Cartesian coordinate to spherical coordinate ($\rho, \theta$ and $\phi$), shown in Fig. 3. Watermark is embedded by modulating $\rho_i$ as it is invariant to similarity and reordering transforms and represents approximately the mesh shape. Embedding in spherical coordinate particularly in $\rho$ is more robust than coordinates $(x, y, z)$. Radial distances of selected vertices are divided into bins and normalized to range $[0, 1]$. One watermark bit is embedded in each bin by modulating radial distance using histogram mapping function as suggested in [11]. Watermark is embedded repeatedly to sustain attacks like simplification and cropping attack. Fig. 4 shows the block diagram of embedding process. Detailed steps for watermark embedding process are provided below.

**Steps of watermark embedding process**

**Step 1:** Vertex selection

Deeper surface vertices are selected for embedding having mean curvature less than zero. Mean curvature ($C_m$) of vertices are calculated using method suggested by M. Mayer et al. [27]. $C_m$ is calculated using following equation.

$$C_m = (\sum_{i=1}^{n} \theta(v_{i,j})) \left(\frac{1}{3} \sum_{k=1}^{n_2} A_k\right)$$

$\theta (v_{i,j})$ denotes the angles between the normal of the two adjacent triangles, and $A_k (k=1, 2, ..., n2)$

$$\sum_{i=1}^{n} \theta(v_{i,j})$$

shows grayscale representation of $C_m$ for bunny model, the darker area represents deeper vertices ($C_m < 0$) used for inserting the watermark, and the lighter area shows superficial vertices ($C_m > 0$).

Fig. 2. Grayscale Representation of Mean Curvature in Bunny 3-D Model.
Step 4: Convert Cartesian coordinates to spherical coordinates

Cartesian coordinates of selected vertices \( v_i \) are converted into spherical coordinates (\( \rho_i, \theta_i \) and \( \varphi_i \)) using given equations:

\[
\rho_i = ((x_i - x_g)^2 + (y_i - y_g)^2 + (z_i - z_g)^2)^{1/2}
\]
\[
\theta_i = \tan^{-1}(\frac{(y_i - y_g)^2}{(x_i - x_g)^2})
\]
\[
\varphi_i = \cos^{-1}(\frac{(z_i - z_g)}{\rho_i})
\]
where \( i \) is the \( i^{th} \) selected vertex and \( (x_g,y_g,z_g) \) are the mean value of all vertices of 3-D model. For the proposed embedding scheme only radial distance \( \rho_i \) of selected vertices are modified while keeping the other two coordinates i.e. \( \varphi_i \) and \( \theta_i \) unaltered.

Step 3: Bin formation

Bins are formed by dividing \( \rho_i \) having equal range with respect to their values. One watermark bit is embedded in each bin. As watermark is embedded repeatedly, number of bins \( N \) is according to the watermark length \( L \) and number of repetitions. Bins are divided by using following equations:

\[
B_n = \{\rho_{n,j}\rho_{\text{min}}
+ ((\rho_{\text{max}} - \rho_{\text{min}})/N).n < \rho_i < \rho_{\text{min}}
+ ((\rho_{\text{max}} - \rho_{\text{min}})/N).(n + 1)\}
\]

where \( 0 \leq n \leq N-1, 0 \leq j \leq M_n-1, M_n \) is the number of radial distance of \( n^{th} \) bin and \( \rho_{n,j} \) is the \( j^{th} \) radial distance of \( n^{th} \) bin. \( \rho_{\text{max}} \) and \( \rho_{\text{min}} \) are the maximum and minimum radial distance of the selected vertices in the mesh model.

Step 4: Bin normalization

Radial distances in the \( n^{th} \) bin are normalized in the range of [0,1] using following equation:

\[
\rho'_{n,j} = (\rho_{n,j} - \min_{\rho_{n,j} \in B_n}\rho_{n,j})
/ (\max_{\rho_{n,j} \in B_n}\rho_{n,j} - \min_{\rho_{n,j} \in B_n}\rho_{n,j})
\]

where \( \rho'_{n,j} \) is the normalized value of \( \rho_{n,j} \), maximum and minimum radial distance of \( n^{th} \) bin is represented by \(\max_{\rho_{n,j} \in B_n}\rho_{n,j}\) and \(\min_{\rho_{n,j} \in B_n}\rho_{n,j}\) respectively.

Step 5: Watermark embedding

One watermark bit is inserted per bin by transforming the radial distances using histogram mapping function as suggested in [11]. The mapping function is defined as Eq 10.

\[
Y = X^\beta \text{ for } 0 < \beta < \infty, \beta \in \mathbb{R}
\]

where \( Y \) is the transformed variable and the parameter \( \beta \) is a real value. Watermark bit ‘1’, is embedded by transforming \( \rho \) to shift the mean by a factor \( \beta \) and for watermark bit ‘0’ shift mean by factor \( -\beta \). Mean of every bin \( \mu_{n,j} \) is shifted by following:

\[
\mu_{n} = \begin{cases} 
\frac{1}{2} + \alpha & \text{if } W = 1 \\
\frac{1}{2} - \alpha & \text{if } W = 0 
\end{cases}
\]

where \( \alpha \) is the watermarking strength to determine robustness and imperceptibility. Mean is shifted by transforming each radial distance using Eq. 10. An iterative algorithm is used to determine the value of parameter \( \beta \) such that the value of modified radial distance exists within the range otherwise it may lead to serious distortions. Algorithm 1. shows the iterative process for determining the value of \( \beta \) and embed watermark bits.

Step 6: Inverse Normalization

Transformed values of every bin are converted back to original range using following equation:

\[
\rho'_{n,j} = (\rho_{n,j} = \max_{\rho_{n,j} \in B_n}\rho_{n,j})
- \min_{\rho_{n,j} \in B_n}\rho_{n,j})
\]

\[
+ \min_{\rho_{n,j} \in B_n}\rho_{n,j})
\]

Step 7: Conversion of spherical coordinates Cartesian to coordinates:

After completing embedding process, arrange and combine all the bins and convert spherical coordinates to Cartesian coordinates using following relation:

\[
x_i' = \rho'_i \cos\theta_i \sin\varphi_i + x_g
\]
\[
y_i' = \rho'_i \sin\theta_i \sin\varphi_i + y_g
\]
\[
z_i' = \rho'_i \cos\varphi_i + z_g
\]

where \( (x_i',y_i',z_i') \) is the vertex Cartesian coordinate of watermarked model and \( \rho'_i \) is the watermarked radial distance, \( \theta_i, \varphi_i \) and center of gravity are same as obtained from Eq. 6 and Eq. 7. Finally, we get the watermarked 3-D mesh.

C. Watermark Extraction

Watermark extraction is just reverse process of watermark embedding. Vertices of watermarked mesh are selected and converted from Cartesian to spherical coordinates in the same way as in the embedding process. Radial distance is divided into bins and normalized using Eq. 8 and Eq.
Algorithm 1: Iterative algorithm to determine the value of parameter $\beta$ and embed watermark bits in bins.

**Require:** Normalized bins

**Ensure:** Watermarked bins

STATE Initialize $\beta=1$ and $\delta\beta=0.001$

1: Transform each normalized radial distance in the bin using Eq. 10.
2: Calculate mean of transformed elements of bin using following equation:
   \[
   \mu_n = \frac{1}{M_n} \sum_{j=0}^{M_n-1} \rho'_{n,j}
   \]
3: if $W=1$ then
4:      if mean$(1/2)+\alpha$ then
5:          $\beta = \beta - \delta\beta$
6:      end if
7:  Goto line number 2.
8: end if
9: else if $W=0$ then
10:   if mean$(1/2)-\alpha$ then
11:      $\beta = \beta + \delta\beta$
12:   end if
13:  Goto line number 2.
14: end if
15: Watermarked and transformed bin is obtained

9. Then mean of each bin is compared with the reference value to extract the watermark. Original mesh is not required for extraction therefore extraction process is blind. More than one, independent watermark sequences are extracted from the bins. These watermark sequences are compared and according to the majority, the final watermark sequence is determined. It acts as error correction mechanism. For example, if $i^{th}$ bit of extracted watermarks has majority of 1 then the final watermark bit is 1; otherwise 0. Finally, the extracted watermark $W'$ is obtained. Fig. 5 shows the block diagram of extraction process. Steps for watermark extraction are explained below.

**Steps of watermark extraction process**

Step 1: Vertices are selected and converted into spherical coordinates as done in step 1 and 2 of watermark embedding process.

Step 2: Bins of radial distance are formed and normalized in the fashion similar to step 3 and 4 of embedding process.

Step 3: Mean is calculated for individual bin and compared with the reference value to extract watermark bits. One bit is extracted from each bin using
following relation:
\[ w'_{n} = \begin{cases} 1 & \text{if } \mu'_{n} > 1/2 \\ 0 & \text{if } \mu'_{n} < 1/2 \end{cases} \]  
(16)

Where \( w'_{n} \) is the extracted watermark bit from \( n^{th} \), \( \mu'_{n} \) is the mean of each bin.

Step 4: More than one watermark sequences are extracted.

Watermark sequences are compared and the final watermark is decided on the basis of majority. Thus, we get 128-bit extracted watermark sequence \( (W') \).

IV. EXPERIMENTAL RESULT AND DISCUSSION

This section presents experimental results to demonstrate the performance of proposed 3-D mesh watermarking scheme. Experiments were conducted to examine the robustness and imperceptibility of the proposed scheme using MATLAB on a PC with Intel core2 Duo, 3.00 GHz, 2GB RAM. Five different 3-D models i.e. Bunny, Armadillo, Dragon, Hand and Venus as shown in Fig. 6 were taken as test data set. These models represent variation of shapes and surfaces. Characteristics of 3-D test models are provided in Table II. Fig. 7 shows the watermarked mesh. For experimental evaluation, 128-bit binary watermark sequence, generated by the watermark generation function, defined as

\[ MOS = \sum_{y=1}^{N} S_{xy} \]  
(17)

Where \( MOS_{x} \) is the mean opinion score of \( x^{th} \) model, \( N \) is the number of observer and \( S_{xy} \) is the score for \( x^{th} \) model by \( y^{th} \) observer.

Maximum root mean square error (MRMS) and Hausdorff distance (HD) are used as metric for objective analysis. MRMS measures surface-to-surface distance between the two meshes. Smaller MRMS value indicates better visual quality. MRMS is calculated by following equation:

\[ MRMS = \max(\text{err}_{RMS}(M,M') \text{, err}_{RMS}(M',M)) \]  
(18)

where \( \text{err}_{RMS} \) is the root mean square error. \( \text{err}_{RMS} \) from one mesh \( M \) to another mesh \( M' \) is given by:

\[ \text{err}_{RMS}(M,M') = \left(1/|M|\right)^{1/2} \int_{p \in M} d(p,M')^{2} dM \]  
(19)

where \( p \) is a point on the surface of mesh \( M \), \( |M| \) is the area of \( M \) and \( d(p,M') \) is the point-to-surface distance between \( p \) and \( M \).

HD is the maximum distance of a set to the nearest point in the other set of object. HD from set A to set B is a max-min function, defined as

\[ HD(O,O'w) = \max \{ \min \{ d(a,b) \} \} \]  
(20)

where, \( a \) and \( b \) are points of original set \( O \) and watermark object \( O'w \) respectively, and \( d(a,b) \) is any metric between these points.

To study the imperceptibility and distortion in watermarked model, both subjective and objective evaluation is done in terms of MRMS, MOS and HD and the results are tabulated in Table III. Subjectively, analysis of original and watermarked models reveals that there is no significant visual distortion is induced. Table III shows that MOS is close to 10 in all models which indicate that no significant perceivable distortion is caused by the proposed scheme. It can be observed from Table III that MRMS and HD values are also very low. Hence, the proposed scheme satisfies both the perceptual and the geometric quality in the 3-D Mesh watermarking.

B. Robustness Evaluation

Robustness is another important requirements of 3-D mesh watermarking scheme. Robustness of proposed scheme is evaluated under different common attacks like, smoothing attack, cropping attack, noise addition, quantization, similarity transformation, simplification attack and scaling on Z axis. Simulation of noise addition and scaling attack is done in MATLAB whereas smoothing, simplification and cropping attacks is simulated by Meshlab open source [28]. Also, the robustness of the proposed scheme is evaluated under attacks of Meshbenchmark tool. Correlation coefficient and BER is used to measure the robustness. It measures the similarity between embedded and extracted watermark and can be express as following equation:

<table>
<thead>
<tr>
<th>Object models</th>
<th>Bunny</th>
<th>Armadillo</th>
<th>Dragon</th>
<th>Hand</th>
<th>Venus</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRMS (×10³)</td>
<td>0.243</td>
<td>0.235</td>
<td>0.283</td>
<td>0.271</td>
<td>0.240</td>
</tr>
<tr>
<td>MOS</td>
<td>9.95</td>
<td>9.88</td>
<td>9.49</td>
<td>9.91</td>
<td>9.84</td>
</tr>
<tr>
<td>Hausdorff Distance</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
</tr>
</tbody>
</table>
Fig. 5. Block Diagram of Extraction Process.

Fig. 6. Original 3-D Mesh Models.

Fig. 7. Watermarked 3-D Mesh Models.
where \( W \) and \( W' \) are embedded and extracted watermark respectively. \( \bar{w} \) and \( \bar{w}' \) are the mean of embedded and extracted watermark, \( w_i \) and \( w'_i \) are the \( i \)th watermark bit of embedded and extracted watermark, respectively. Correlation ranges in \([-1, 1]\) and the ideal value is 1.

When the watermark is extracted from watermarked model without any attack, the corr value is 0.9992 quite close to the ideal value. Watermark embedding process effects the correct bin division in the watermark extraction process therefore ideal value of corr could not be achieved.

1) Robustness Against Smoothing Attack: Smoothing is commonly used to eliminate the noise introduced during the mesh generation process. There are many smoothing algorithms out of which Laplacian smoothing algorithm proposed in [29] is commonly used. Therefore, in existing literature majority of the 3-D watermarking schemes test robustness of their scheme under Laplacian smoothing attack. It is an iterative process where vertex position is calculated by averaging its neighbours and the weight is determined according to its local connectivity. The results of Laplacian smoothing with different number of iteration ranging from 10-50 and keeping deformation factor \( \lambda = 0.03 \) is presented in Table VI. From Table VI, it can be observed that the corr value for all models is appreciable up to 30 iterations. Also, the extracted watermark is distinguishable for less than 50 iterations as the normalized correlation is above the threshold value. Hence, the proposed scheme has fair robustness under smoothing attack.

2) Robustness Against Cropping: Cropping attack deletes vertices from one or more parts of a 3-D mesh model. It destroys the watermark information from the cropped region but watermark can be validated from the remaining portion. Cropping attack does not destroy all watermarked vertices because watermark bits are scattered uniformly. In the proposed scheme watermark is repetitively embedded to resist cropping attack. From the Table IV it can be observed that corr value for 10%, 20% and 25% cropping are more than 0.92, 0.83 and 0.75 indicating that extracted watermark is distinguishable. Thus, the proposed scheme is robust against cropping attack, because for all models corr value is more than the threshold.

### TABLE IV. CORR VALUES FOR 3-D MODELS UNDER CROPPING ATTACK

<table>
<thead>
<tr>
<th>Cropping %</th>
<th>Bunny</th>
<th>Armadillo</th>
<th>Dragon</th>
<th>Hand</th>
<th>Venus</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.9235</td>
<td>0.9556</td>
<td>0.9375</td>
<td>0.9607</td>
<td>0.9451</td>
</tr>
<tr>
<td>20</td>
<td>0.8371</td>
<td>0.8593</td>
<td>0.8299</td>
<td>0.8788</td>
<td>0.8488</td>
</tr>
<tr>
<td>25</td>
<td>0.7598</td>
<td>0.7827</td>
<td>0.7621</td>
<td>0.7494</td>
<td>0.7399</td>
</tr>
</tbody>
</table>

3) Robustness Against Simplification Attack: Mesh simplification is commonly used operation particularly for compression. A highly detailed mesh can be simplified or conversely increase the complexity of simplified mesh. Table V shows Corr values for varying simplification ratio. The proposed scheme shows satisfactory resistance towards simplification attack.

### TABLE V. CORR VALUES FOR 3-D MODELS UNDER SIMPLIFICATION ATTACK

<table>
<thead>
<tr>
<th>Simplification vertex reduction ratio (%)</th>
<th>Bunny</th>
<th>Armadillo</th>
<th>Dragon</th>
<th>Hand</th>
<th>Venus</th>
</tr>
</thead>
<tbody>
<tr>
<td>70</td>
<td>0.9046</td>
<td>0.8571</td>
<td>0.8735</td>
<td>0.8579</td>
<td>0.8699</td>
</tr>
<tr>
<td>90</td>
<td>0.8136</td>
<td>0.7889</td>
<td>0.8025</td>
<td>0.7792</td>
<td>0.7964</td>
</tr>
</tbody>
</table>

### TABLE VI. CORR VALUES FOR 3-D MODELS UNDER LAPLACIAN SMOOTHING ATTACK (\( \lambda = 0.03 \))

<table>
<thead>
<tr>
<th>Iterations</th>
<th>Bunny</th>
<th>Armadillo</th>
<th>Dragon</th>
<th>Hand</th>
<th>Venus</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.9386</td>
<td>0.9025</td>
<td>0.9144</td>
<td>0.9247</td>
<td>0.9197</td>
</tr>
<tr>
<td>30</td>
<td>0.8602</td>
<td>0.8703</td>
<td>0.8463</td>
<td>0.8836</td>
<td>0.8961</td>
</tr>
<tr>
<td>50</td>
<td>0.6173</td>
<td>0.6899</td>
<td>0.5894</td>
<td>0.6391</td>
<td>0.5939</td>
</tr>
</tbody>
</table>

4) Robustness Against Random Additive Noise: Noise attack is very common during data transmission and mesh object operation like read/write. Pseudo random noise (amplitude \( \epsilon \ 0.05\%, 0.25\%, 0.5\% \)) is added on vertex to distort the watermark. Experimental results of random additive noise attacks in Table VII shows that proposed scheme shows good robustness even at high noise ratio of 0.5%.

### TABLE VII. CORR VALUES FOR 3-D MODELS UNDER ADDITIVE NOISE ATTACK

<table>
<thead>
<tr>
<th>Noise %</th>
<th>Bunny</th>
<th>Armadillo</th>
<th>Dragon</th>
<th>Hand</th>
<th>Venus</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.9883</td>
<td>0.9497</td>
<td>0.9578</td>
<td>0.9479</td>
<td>0.9397</td>
</tr>
<tr>
<td>0.3</td>
<td>0.9146</td>
<td>0.8692</td>
<td>0.8749</td>
<td>0.8491</td>
<td>0.8739</td>
</tr>
<tr>
<td>0.5</td>
<td>0.8139</td>
<td>0.7616</td>
<td>0.7509</td>
<td>0.7787</td>
<td>0.7989</td>
</tr>
</tbody>
</table>

5) Robustness Against Quantization Attack: Quantization is lossy compression method. We have evaluated robustness of our scheme against different quantization rate. Vertex coordinate is represented with 7, 8 and 9 bits. Table VIII shows the corr values against different quantization attack. Proposed method is robust for 9-bit and 8-bit quantization attack.

### TABLE VIII. CORR VALUES FOR 3-D MODELS UNDER ADDITIVE QUANTIZATION ATTACK

<table>
<thead>
<tr>
<th>Quantization %</th>
<th>Bunny</th>
<th>Armadillo</th>
<th>Dragon</th>
<th>Hand</th>
<th>Venus</th>
</tr>
</thead>
<tbody>
<tr>
<td>9 bits</td>
<td>0.9983</td>
<td>0.9448</td>
<td>0.9753</td>
<td>0.9488</td>
<td>0.9961</td>
</tr>
<tr>
<td>8 bits</td>
<td>0.9461</td>
<td>0.9121</td>
<td>0.8942</td>
<td>0.8813</td>
<td>0.8498</td>
</tr>
<tr>
<td>7 bits</td>
<td>0.6191</td>
<td>0.7114</td>
<td>0.5529</td>
<td>0.6771</td>
<td>0.7389</td>
</tr>
</tbody>
</table>

6) Robustness Against Similarity Transform: It is desirable for any 3-D watermarking scheme to be robust against similarity transform as it includes rotation, translation, and uniform scaling attack. Table IX presents corr values under similarity transformation. It can be observed from Table IX that the proposed watermarking scheme is robust against this attack for all 3-D model used for evaluation.

7) Robustness Against Scaling Attack on Z-axis: Uniform scaling attack on Z-axis having scaling factor 1.25 is applied on 3-D models. Results are tabulated in following Table X. Proposed scheme is fairly robust against scaling attack.
The computational cost is $O(nV)$ where $n$ is the number of iterations.

Based on the above analysis, we conclude that the proposed watermarking scheme has linear time complexity as:

$$O(3V) + O(V) + O(nV) = O(nV)$$  \hspace{1cm} (22)$$

The average number of iteration and computation time required for determining $\beta$ for each bin is presented in Table XII. The total time required for watermark embedding ranges from 0.6s to 200s subjected to mesh size.

### D. Comparative Analysis

To further validate the performance of the proposed scheme, a comparative study is done with existing 3-D mesh watermarking schemes in terms of imperceptibility, robustness and security. Comparison has been done with spatial domain schemes proposed by Zhan et al. [10], Li et al. [13], and transform domain scheme proposed by Hamidi et al. [17]. In the scheme proposed by Zhan et al. [10], watermark bits are interleaved in bins formed on the basis of vertex curvature. Similarly, Li et al. [13] suggested to form bins of eigen values (distance between vertex to the model center) based on improved vertex grouping. It uses piecewise mapping function to insert watermark bits in the bins. Both the watermarking schemes are in spatial domain and blind. They show improved imperceptibility and robustness than [11], but none of the them have taken security requirement of watermarking in account. Hamid et al. [17] have proposed a to blind watermarking scheme in transform domain, where watermark is embedded in quantified wavelet coefficients using QIM based on 3-D mesh saliency. Though the scheme achieves good imperceptibility and robustness but, it is not able to resist cropping attack. They have used key, but its effectiveness in providing security is debatable. Schemes in comparison, assert to have better performance than the state-of-the-art schemes in existing literature.

Imperceptibility of the proposed scheme is compared with the other counterparts [10], [13], [17] in terms of MRMS value. Fig. 8 shows the MRMS value for three watermarked 3-D mesh (without attack) i.e. Bunny, Dragon and Venus. MRMS value of the proposed scheme is lower than [10], [17] and quite equal to [13]. Hence, it can be inferred that the proposed scheme outperforms [10], [17] and is at par with [13] in terms of imperceptibility. Higher imperceptibility of the proposed scheme can be attributed to the embedding of watermark bits in deeper vertices.

Further, comparison of performance under different attacks is studied in terms of Corr and MRMS value. Table XIII shows comparison of robustness and imperceptibility of the proposed scheme with Zhan et al. scheme [10]. It can be observed from

<table>
<thead>
<tr>
<th>Attacks</th>
<th>BER (Perceptual)</th>
<th>BER (Geometric)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Similarity tranform</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>Noise A = 0.10%</td>
<td>0.01</td>
<td>0.04</td>
</tr>
<tr>
<td>Noise A = 0.30%</td>
<td>0.08</td>
<td>0.12</td>
</tr>
<tr>
<td>Noise A = 0.50%</td>
<td>0.19</td>
<td>0.25</td>
</tr>
<tr>
<td>Smoothing $N_{cfe}$ = 10</td>
<td>0.13</td>
<td>0.22</td>
</tr>
<tr>
<td>Smoothing $N_{cfe}$ = 30</td>
<td>0.29</td>
<td>0.37</td>
</tr>
<tr>
<td>Smoothing $N_{cfe}$ = 50</td>
<td>0.39</td>
<td>0.42</td>
</tr>
<tr>
<td>Quantization $R$ = 9</td>
<td>0.09</td>
<td>0.16</td>
</tr>
<tr>
<td>Quantization $R$ = 8</td>
<td>0.18</td>
<td>0.21</td>
</tr>
<tr>
<td>Quantization $R$ = 7</td>
<td>0.42</td>
<td>0.51</td>
</tr>
<tr>
<td>Simplification $E_{cfe}$ = 10%</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>Simplification $E_{cfe}$ = 30%</td>
<td>0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>Simplification $E_{cfe}$ = 50%</td>
<td>0.06</td>
<td>0.17</td>
</tr>
<tr>
<td>Cropping $V_c$ = 10%</td>
<td>0.51</td>
<td>0.53</td>
</tr>
<tr>
<td>Cropping $V_c$ = 30%</td>
<td>0.55</td>
<td>0.56</td>
</tr>
<tr>
<td>Subdivision Midpoint</td>
<td>0.01</td>
<td>0.03</td>
</tr>
</tbody>
</table>
Table XIII that the proposed scheme has higher robustness and imperceptibility under smoothing, quantization and cropping attack for majority of models in comparison. In simplification attack, correlation value of the proposed scheme is lesser than [10] but higher than the threshold. Under these attacks, MRMS values are comparatively higher for the proposed scheme indicating higher imperceptibility. And, the proposed scheme has vying performance against simplification, noise and scaling attack. Therefore, it can be said that the proposed scheme outperforms in most of the common attacks than Zhan et al. scheme [10]. Similarly, performance comparison of the proposed scheme has been done with Li et al. scheme [13] in Table XIV. It can be observed from Table XIV that the proposed scheme outperforms [13] in terms of imperceptibility under all attacks. The proposed scheme performs better under smoothing and quantization attacks. Though, the robustness of the Li et al. scheme [13] is higher under noise attack than the proposed scheme but lags behind in imperceptibility. Thus, we can deduce that the proposed scheme outdo [13]. Performance of the proposed scheme is also compared with Hamidi et al. scheme [17] under different attacks for Bunny model and the results have been tabulated in Table XV. It can be observed that the proposed scheme is more robust than Hamidi et al. scheme [17] under common attacks except noise attack.

An overall comparison of robustness for the proposed scheme and counterpart schemes [10], [13], [17] under smoothing, quantization and noise is presented in Fig. 9, Fig. 10 and Fig. 11 respectively. Fig. 9 shows that under smoothing attack, the proposed scheme has higher robustness than [10], [13] but lags behind [17] when number of iterations are increases more than 40. Under quantization attack the proposed scheme has

![Fig. 8. MRMS Value for Watermarked Bunny, Dragon and Venus for Schemes Proposed in [10], [13], [17].](image-url)
better performance than [10], [13] and is at par with [17] as shown in Fig. 10. The proposed scheme is more robust than [13] under noise attack and passable in comparison to schemes in [10], [17]. Similarly, an overall comparison of imperceptibility is presented in Fig. 12, Fig. 13 and Fig. 14. From these figures, it can be observed that the proposed scheme has appreciably lower MRMS values than [10], [13], [17] under smoothing, quantization and noise. This indicates that the proposed scheme has higher imperceptibility. Hence, from the comparative study we can conclude that the proposed scheme shows overall improved performance than [10], [13], [17] in terms of imperceptibility, robustness and has higher security.

V. CONCLUSION

In this paper, a blind, robust, and secure 3-D mesh watermarking scheme is proposed that embeds watermark bits by modulating the radial distance distribution of deeper vertices using histogram mapping function. A unique watermark sequence is generated using MD5 hash function according to the grayscale watermark/logo to ensure higher security. Experimental evaluation of watermarked 3-D models reveals that no significant distortion is induced by watermarking process hence, the visual quality is good. Also, that the proposed scheme is robust against various common attacks like smoothing, additive noise, quantization, similarity transformation, cropping and simplification. Comparative analysis reflects that the proposed scheme outperforms the existing schemes and may find potential solution to the copyright protection applications. Since watermark bits are embedded in deeper vertices of 3-D mesh, this scheme may not be effective for models with only positive mean curvature or flat models. This can be seen as a scope for future work. In future we wish to improve robustness of the proposed scheme using hybrid transforms.

REFERENCES


