

PAPER • OPEN ACCESS

Analysis of the ratio expansion watermarking scheme and its improvement

To cite this article: Keming Dong and Xiaohan Yu 2020 *J. Phys.: Conf. Ser.* **1570** 012059

View the [article online](#) for updates and enhancements.

You may also like

- [A content-based digital image watermarking scheme resistant to local geometric distortions](#)
Hong-ying Yang, Li-li Chen and Xiang-yang Wang
- [A reversible image watermarking scheme based on a new modulation mode of DCT coefficients](#)
A P Bahrushin, G I Bahrushina and R I Bazhenov
- [Metric and classical fidelity uncertainty relations for random unitary matrices](#)
Radosaw Adamczak



ECS
The
Electrochemical
Society
Advancing solid state &
electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research

Analysis of the ratio expansion watermarking scheme and its improvement

Keming Dong¹ and Xiaohan Yu^{2*}

¹ Yunnan Universities Research Center for Data Operation and Management Engineering, Yunnan University of Finance and Economics, Kunming, Yunnan, 650221, China

²School of Information and Electronic Engineering, Zhejiang Gongshang University, HangZhou, ZheJiang, 310018, China

*Corresponding author's e-mail: yuxiaohan188@126.com

Abstract. Ratio expansion is a recently proposed, remarkable method for reversible data hiding in 3D meshes. The capacity and distortion are the two most common performance measures for data hiding. This paper analyses the ratio expansion method, and a novel mathematical framework is introduced and described, which proves that the magnitude of the distortion depends on three parameters. By appropriately selecting two of these parameters and a factor called “reference value”, the proposed method achieves reduction of the distortion. Another novel coding and decoding phrase is also proposed to get higher capacity. The experimental results show that this method improves the embedding capacity significantly without affecting the visual quality of 3D mesh model.

1. Introduction

Due to the recent advances in computing power and transmission bandwidth, three-dimensional (3D) digital content becomes more and more popular in many application domains such as video games, computer animation, virtual reality. As a result, the need for copy protection, authentication and integrity checking of the 3D content emerges. Watermarking is an appropriate approach to address this need. 2D image watermarking is a well-established and studied field for recent years. Difference expansion [10], [25] and histogram shifting [17] methods are nowadays combined [21], [23], [24], [32], [33] creating a mainstream in the 2D reversible watermarking domain. However, the methods that have been proposed so far cannot be directly applied to 3D models due to their numerous alternative representation structures (point clouds, polygonal meshes, parametric surfaces, etc.) and presentation models (texture, reflective characteristics, etc). For example, Luo et al. [16] apply their algorithms on point cloud models, while Garcia and Dugelay [8] work on textured-based 3D objects.

In general, watermarking schemes can be categorized into irreversible and reversible ones. Irreversible watermarking schemes recover only embedded messages, while the reversible ones recover both distorted host signals and embedded messages. Thus, reversible watermarking schemes are also called invertible data hiding, lossless data hiding, or distortion-free data hiding.

The first 3D watermarking algorithm applied on a 3D mesh model was published by Ohbuchi et al. [18]. Since then, numerous algorithms have been developed in the irreversible watermarking domain [1], [4], [5], [6], [13], [19], [27], [30], [31], [34] which aimed at designing robust watermarking algorithms against all possible attacks. Most of these embedding techniques modify, and hence distort,



the host signal in order to insert additional information. Often, this embedding distortion is small, yet irreversible. In many applications, the loss of host signal fidelity is not prohibitive as long as original and modified signals (in our case 3D models) are perceptually equivalent. However, in some application domains, such as medical imaging, military imaging and law enforcement, a slight modification can lead to a significant difference in the final decision-making process. Thus, it may affect the right decision and lead to legal problems. The latter implies that reversible (lossless) data embedding techniques are needed where the host signal must be perfectly recovered.

According to different embedding domains, the reversible watermarking schemes can be classified into spatial-domain-based, compressed-domain-based and transform-domain-based methods. In spatial-domain-based methods [7], [28], [29], the task of data embedding is to modify the vertex coordinates, edge connections, face slopes, and so on. These schemes usually have a low computational complexity. The compressed-domain-based methods [15], [22] are used for embedding data with certain compression techniques involved, e.g., vector quantization. In addition, some of these methods are designed for compressed content of 3D models. Their advantage is that they hide data without decompressing the host model. In transform-domain based methods [16], the original model is transformed into a certain transform domain (i.e., wavelet transform domain in their approach) and then data are embedded in the transformed coefficients.

One of the first reversible watermarking methods for 3D mesh models was published in 2006 by Sun et al. [22]. This method utilized a vector quantization compression technique and the space saved due to the compression was used to place the watermark. The performance of this approach was related to the efficiency and the distortion caused by the compression method.

Jhou et al. [9] proposed a method based on histogram shifting for 3D reversible watermarking. They firstly calculated the center of the vertices of the model and all vertices were sorted by their distance to the center, in descending order. The last m bits of the coordinate value of every vertex were used to generate a histogram. Then, the watermark embedding was taken place as a normal histogram shifting method for images.

The prediction error expansion (PEE) method [29] predicts a vertex position by calculating the centroid of its traversed neighbors, i.e., the vertices directly connected and having been previously traversed. By comparing the predicted and real positions, the prediction error can be obtained and expanded to embed one-bit value in each modified coordinate. In order to reduce the distortion, prediction errors with magnitude larger than an adaptively chosen threshold are not being expanded. In another relevant approach called ratio expansion (RE) [12], the distance between an anchor vertex and its adjacent vertices is divided by a reference value to calculate a ratio. Every ratio is used to achieve reversible watermarking by exploiting a modulo arithmetic operation. These two approaches show superior performance compared to the previously mentioned methods and will be used in this paper as a reference for evaluating the proposed scheme. In general, RE outperforms PEE in terms of perceptual quality but has smaller embedding capacity.

In this paper, we propose an improved RE-based method that can increase the embedding capacity and decrease the distortion. The former is achieved by proposing a novel encoding and decoding phrase to get more vertices to embedding. Concerning the latter, mathematical formulae are derived, and parameters that affect the distortion are calculated and experimentally verified. Extensive experimental results performed prove that the proposed scheme can significantly improve the performance of the existing RE [12] methods in terms of both the distortion and the capacity.

The rest of this paper is organized as follows: Section 2 briefly describes the existing ratio expansion algorithms. In Section 3, the improved ratio expansion method is introduced. Then, the expected distortion of the ratio expansion algorithm is estimated, and experiments are conducted to verify it. Experimental results are given in Section 5. Finally, conclusions are drawn in Section 6.

2. The ratio expansion method

In 2010, Ji et al. [12] proposed a RE scheme for reversible watermarking. In their scheme, one “anchor” (or “center”) vertex v_0 and its K adjacent vertices v_i can be considered as an embedding unit. One of

the anchor's adjacent vertices is selected to be the reference vertex v_l , if it has the smallest index among all neighbouring vertices in an embedding unit. A ratio R of the L_2 -distance D from the center vertex to an adjacent vertex to the distance from the center vertex to the reference vertex is then computed as:

$$R(v_0, v_i) = \frac{D(v_0, v_i)}{D(v_0, v_1)}, \quad i = 2, 3, \dots, K. \quad (1)$$

Every ratio can be expanded to embed n bits of data (in general, n is at least 1). For each embedding unit with M vertices, the center vertex and a reference vertex should not be modified, thus $M-2$ vertices can be expanded using the RE method. Therefore, the maximum embedding capacity is $n \cdot (M-2)$ bits. However, every embedding unit should not overlap each other, leading to a not so large embedding capacity of the RE method.

Figure 1 shows two valid embedding units. The anchor vertex v_l is connected to six neighboring vertices: v_2, v_3, v_4, v_5, v_9 , and v_{11} . In this embedding unit, M equals 7 since there are six vertices around the anchor vertex. Thus, we can hide $5n$ bits of data. The vertex v_2 becomes v_0 as a reference as it has the smallest index.

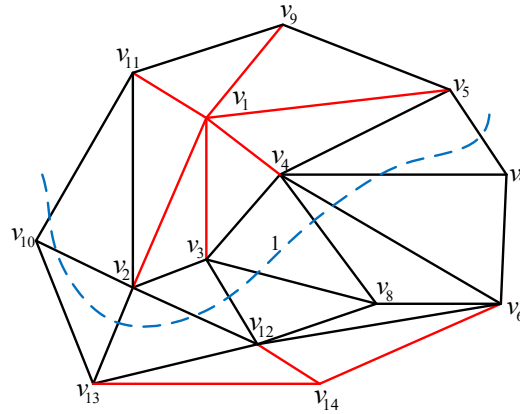


Figure 1. Possible anchor vertices of the RE method.

The vertex v_2 cannot be an anchor to form embedding unit since it is a member of the existing unit. For the same reason, v_3, v_4 and v_5 . The vertex v_6 is not a member of any existing unit, however, its adjacent v_4 is the member of the existing unit, so it should be skipped too. For the same reason, we skip vertices from v_7 to v_{13} . The vertex v_{14} can form an embedding unit with three neighbouring vertices: v_6, v_{12} , and v_{13} . In this embedding unit, M is 3. Even though there are fourteen vertices in figure 1, the embedding capacity is $6n$ (i.e., $(5+1)n$) bits. Each embedding unit is separated by dashed lines depicted in figure 1.

Two parameters are used in RE: n that determines the number of bits to be hidden and m that specifies the embedding position. By assuming that the secret data (watermark) should have k units $W1, W2, \dots, Wk$, then W_i can be one of a 2^n -ary numbers, i.e., $W_i \in \{0, 1, 2, \dots, 2^n - 1\}$. Before expanding R (see equation (1)), we need to know m so as to specify the embedding position in R .

Two distinct steps are followed in the RE method: In the first step, the ratio R is divided by 2^n , and then the quotient is split into two parts according to the position m . The first part is the embedding part R_e , and the second is the recording part R_r , which are computed as follows:

$$R_e = T_m(R/2^n), \quad (2)$$

$$R_r = R/2^n - R_e. \quad (3)$$

where $T_m(\cdot)$ is a function that truncates the part right to the position m . Embedding is achieved by:

$$R'_e = R_e \cdot 2^n + W_i \cdot 10^{-m}, \quad (4)$$

$$R' = R'_e + R_r, \quad (5)$$

where W_i stands for the i^{th} secret data. The anchor vertex v_0 and the reference vertex v_l never expand, while the neighbouring vertex v_i with reference to the center vertex does expand along the direction $\overrightarrow{v_0 v_l}$.

In the extraction phase, R'_e and R_r are calculated as:

$$R'_e = T_m(R'), \quad (6)$$

$$R_r = R' - R'_e, \quad (7)$$

where R' is the ratio from the watermarked 3D meshes. The hidden message (watermark) W_i can be extracted using:

$$W_i = M(R'_e \cdot 10^m, 2^n), \quad (8)$$

where $M(x, y)$ is the remainder of x divided by y , and R'_e is the embedding part of R' . The function $M(x, y)$ is similar to the modulus operation. The model can be recovered through:

$$R = \left[\frac{(R'_e - W_i \cdot 10^{-m})}{2^n} + R_r \right] \cdot 2^n. \quad (9)$$

3. Proposed method

In this section, we firstly introduce the mathematical framework of the distortion caused by ratio expansion embedding. The framework shows that the distortion relates to three parameters, that is, “reference value”, embedding position m and n . Then, the proposed method selects the shortest edge among the edges connected to the center vertex as the reference value to decrease the distortion. Moreover, a new encoding and decoding phase are proposed to improve the capacity.

3.1 Mathematical Expectation of the Distortion for the RE

In [12], one pixel and its all neighbors are considered as an embedding unit, and the neighbor with the smallest index among all neighbors will be selected as the reference. The selection of reference is not so reasonable. In this section, we try to deduce PDF (probability density function) of distortion caused by one ration expansion embedding, and then obtain the mathematical expectation of the distortion. Based on the expectation, we can select the references and config the other parameters.

We firstly review the symbols. R is the ratio before data hiding and R' is the one after data hiding. There are two parameters, m and n . m is the position in R where the data is embedded. The secret unit is embedded to m right to decimal point of R . n determines the ability for one expansion embedding. For example, if m is 4 and R is 1.12345678, then secret number is embedded to the place of number 4; if n is 2, then we can embed one number of “0, 1, 2, 3” to R for one embedding. W_i is the secret number we want to embed, and it can be one of “0, 1, 2, ..., 2^{n-1} ”. R_e and R_r is the embedded part and recording part of R , respectively. The definitions are in equation 3. R'_e is the embedded part of R' , please refer to equation 4. Visualization of these symbols can be found in figure 2.

We still have to define three more symbols. Let d denote the distortion measure caused by an expansion and r denote the remainder after dividing by 2^n , that is:

$$r = \text{mod}(T_m(R) \cdot 10^m, 2^n), \quad (10)$$

where mod is the modulus after division, and $R_{[m+1, \infty)}$ is defined as follows:

$$R_{[m+1, \infty)} = (R - T_m(R)) \cdot 10^m. \quad (11)$$

An example is made. Assume R is 1.12345678, m is 4, n is 2 and W_i is 3. In the ratio expansion, R is firstly divided by 22, and the result is 0.280864195. We can obtain that R_e is 0.2808 and R_r is 0.000064195, then R_e will be multiplied by 2^2 and added by W_i at “position 4” to form R'_e , that is, $0.2808 \times 4 + 3 \times 0.0001$, so R'_e is 1.1235. R_r is appended to R'_e to form R' , that is 1.123564195. In this embedding, r and $R_{[m+1, \infty)}$ can be obtained from R . r is $\text{mod}(T_4(1.12345678) \cdot 10^4, 4)$, which is 2, and $R_{[m+1, \infty)}$ is 0.5678.

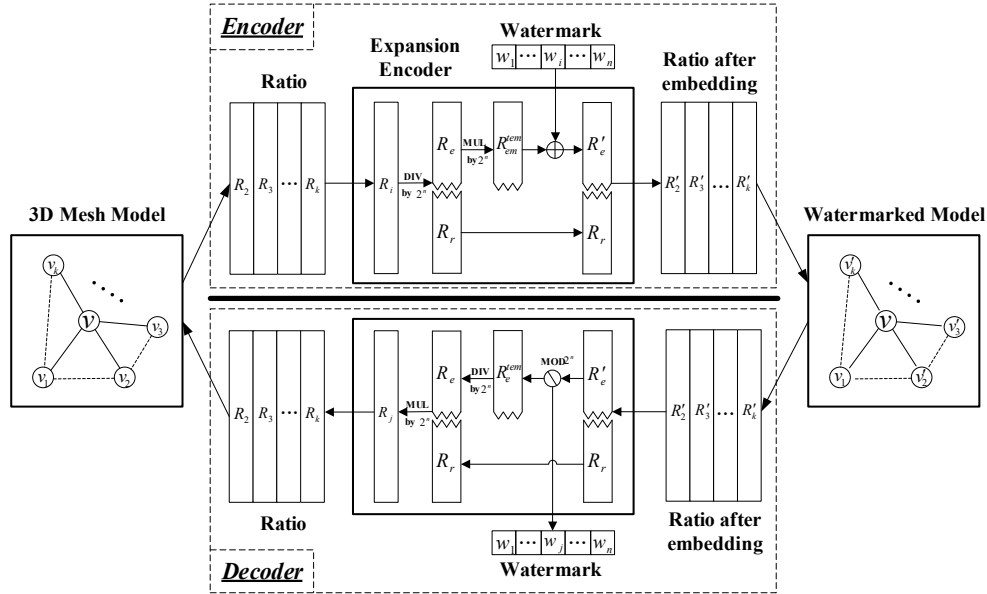


Figure 2. Framework of the ratio expansion scheme.

Our deduction of PDF p_d is based on the following assumptions:

- r and W_i are discrete, uniformly distributed over $\{0, 1, \dots, 2^{n-l}\}$, and $R_{[m+l, \infty)}$ is continuous, uniformly distributed over $[0, 1)$.

- r , W_i and $R_{[m+l, \infty)}$ are mutually independent.

In the following, we firstly deduce function $d(W_i, r, R[m+1, \infty), v_f)$ which is in equation (15), and then compute p_d .

The distortion caused by one expansion embedding is expressed as follows:

$$d = |R' - R| \cdot v_f, \quad (12)$$

where v_f is the reference value. Substituting equations (3) and (4) with equation (12), we obtain

$$d = |R_r \cdot (2^n - 1) - W_i \cdot 10^{-m}| \cdot v_f. \quad (13)$$

Then, according to equation (3), R_r can be computed as

$$R_r = 10^{-m} \cdot (r + R_{[m+1, \infty)}) / 2^n. \quad (14)$$

Finally, we can draw

$$d = 10^{-m} \cdot \left| (r + R_{[m+1, \infty)}) \cdot \frac{2^n - 1}{2^n} - W_i \right| \cdot v_f. \quad (15)$$

Let we define ∇ as follows:

$$\nabla = (r + R_{[m+1, \infty)}) \cdot \frac{2^n - 1}{2^n} - W_i, \quad (16)$$

For the sake of brevity, we replace a_1 , a_2 , and b with W_i , r , and $R_{[m+1, \infty)}$, respectively. The probability density functions p_{a1} and p_{a2} of the random variables a_1 , a_2 can be expressed as $P(x)$ in the following based on the above distributions:

$$p_a(x) = \begin{cases} 1/2^n & x \in \{0, 1, \dots, 2^n - 1\} \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

and b as

$$p_b(x) = \begin{cases} 1 & x \in [0, 1) \\ 0 & \text{otherwise} \end{cases}. \quad (18)$$

Then, the probability density function of $a_2 + b$ can be easily derived based on Lemma 1.

$$p_{a_2+b}(x) = \begin{cases} 1/2^n & x \in [0, 2^n) \\ 0 & \text{otherwise} \end{cases}. \quad (19)$$

Lemma 1. Let X and Y be independent random variables having the respective probability density functions $f_X(x)$ and $f_Y(y)$. Then the probability distribution function $f_Z(z)$ of the random variable $Z = X + Y$ can be given as follows:

$$f_Z(z) = \int_{+\infty}^{-\infty} f_X(x)f_Y(z-x)dx. \quad (20)$$

Proof: see [20].

According to Lemma 2, we can get the probability density function of $\frac{2^n-1}{2^n} \cdot (a_2 + b)$ as

$$p(x) = \begin{cases} 1/(2^n - 1) & x \in [0, 2^n - 1) \\ 0 & \text{otherwise} \end{cases}. \quad (21)$$

Lemma 2. If $X \sim F_X$, then $F_X(x) = \text{Prob}\{X < x\}$ for $\underline{x} < X < \bar{x}$. Let $Y = aX$ for $a\underline{x} < Y < a\bar{x}$; then $F_Y(y) = \text{Prob}\{Y < y\} = \text{Prob}\{aX < y\} = \text{Prob}\{X < y/a\} = F_X(y/a)$. Therefore, $F_Y(y) = F_X(y/a)$ for $a\underline{x} < Y < a\bar{x}$.

Proof: See [20].

Using Lemma 1 again, we can derive the probability density function of $|\nabla|$ as follows:

$$p(x) = \begin{cases} \frac{||x||-2^n}{(2^n-1) \cdot 2^n} & x \in [0, 2^n - 1) \\ 0 & \text{otherwise} \end{cases}. \quad (22)$$

The probability density function figures of $a_1, a_2, b, \frac{2^n-1}{2^n} \cdot (a_2 + b)$ and ∇ are shown in figure 8.

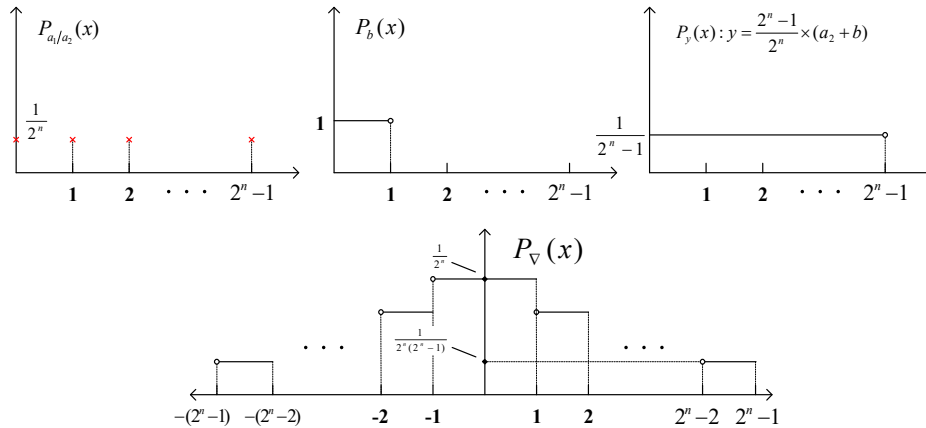


Figure 3. Probability density function of the ratio expansion scheme.

Then, the mathematic expectation of $|\nabla|$ can be computed according to its definition, and the result is given as follows:

$$E(|\nabla|) = \frac{1}{6} \cdot (2^{n+1} - 1). \quad (23)$$

Equation (15) shows $d = 10^{-m} \cdot |\nabla| \cdot v_f$. According to the linearity of the expectation operator, the expectation of d is given as follows:

$$E(d) = \frac{1}{6} \cdot (2^{n+1} - 1) \cdot 10^{-m} \cdot E(v_f). \quad (24)$$

This equation implies that:

- The distortion d is proportional to 10^{-m} and v_f .
- d is an increasing function of n .

3.2 Verification

In order to further verify the conclusions of the previous subsection, we use the Bunny as a test model, choosing different values for the parameters m and n , and D_r . A standard metric, the signal-to-noise ratio (SNR), is used as a performance measure.

The first experiment is conducted to identify the effect of the reference values. In this experiment, the shortest edge among the edges connected with the center will be chosen as the reference for every embedding step. Figure 4 verifies that the method with new reference produces less distortion.

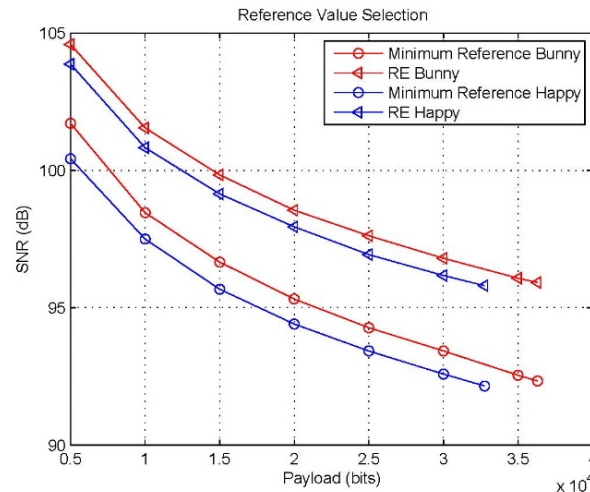


Figure 4. Distortion using different reference values on Bunny, when other parameters are fixed.

In table 1, we report how parameters n and m affect distortion. In this experiment, we embed 3,000 units of secret data into Dragon, Happy, and Bunny with different parameter values of n and m , respectively. Table 1 shows that smaller n and larger m produce smaller distortion. As a result, we select the shortest adjacent edge among the three sides of a triangle as the reference value, and set $m = 2$ and $n = 3$ in all experiments that follow.

Table 1. Distortion using different m or n values.

Model	N				m			
	1	2	3	4	2	3	4	5
Dragon	104.57	96.83	90.10	83.83	76.76	96.75	116.80	136.71
Happen	109.04	101.31	94.61	88.27	81.29	101.27	121.26	141.28
Bunny	109.17	101.34	94.68	88.37	81.35	101.37	121.39	141.43

3.3 Encoding and Decoding process

3.3.1 Encoding Phase. In embedding process, the index of the last embedded vertex (stego-vertex) should be sent to decoder as the header information. The embedding steps are as follows.

Step 1. Set the flags of all vertices to 0 and the set V to null.

Step 2. Traverse each vertex of the model. For each point v_i , set the set V_i to null as initial value. Traverse each adjacent vertex of v_i , and if the flag of the adjacent vertex is 0, add this vertex to V_i . If the flag of v_i is 1 or the cardinality of V_i is less than 2, set V_i to null; else the flags of v_i and the vertices in V_i are all set to 1.

Step 3. For every vertex v_i in V , if V_i is not null set, each distance from v_i to the vertex in V_i is computed. The reference value is the minimum distance, which will respectively divide other distances, and the ratios will be saved.

Step 4. For the results obtained from step 3, we embed watermarks by ratio expansion, and then the ratios are changed.

Step 5. According to the changed distance ratios, relevant points are shifted along certain direction, finally forming a watermarked model.

3.3.2 Decoding Phase. In The restoration of original model and extraction of watermarks is just reverse to the procedure of data embedding. Steps are as follows.

Step 1. Follow the same way as steps 1 and 2 did in embedding. Not only the flag values of all vertices are obtained, but also the set queue V_i is attained, where $i \in \{1, 2, \dots, lv\}$, and lv is the index of the last embedded vertex.

Step 2. The decoding uses the opposite direction of the encoding, from the last embedded vertex to the first vertex. For every vertex V_i , if V_i is not a null set, the ratios are obtained as step 3 does in embedding, and then the flags of v_i and the vertices in V_i are all set to 0; else go to previous vertex, and do step 2 again.

Step 3. For the ratios obtained in step 2, we extract the watermarks, and recover the model by ratio expansion method.

Step 4. Go to previous vertex and then do step 2 again until finishing processing the first vertex.

4. Experiments

In this section, we firstly demonstrate the visual quality of the proposed method, and then we compare our method with the work in [10] in terms of embedding capacity versus distortion. The test models are shown in figure 5. In the experiments, the parameter values $m = 3$ and $n = 2$ are used.

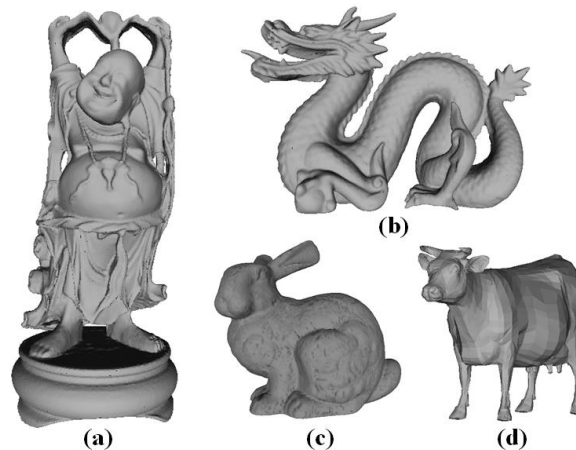


Figure 5. Test models: (a) Happy, (b) Dragon, (c)Bunny, (d) Cow.

Figure 6 depicts the achieved embedding capacity versus distortion of the proposed method compared against the RE method. It is obvious that the proposed method clearly outperforms the RE method.

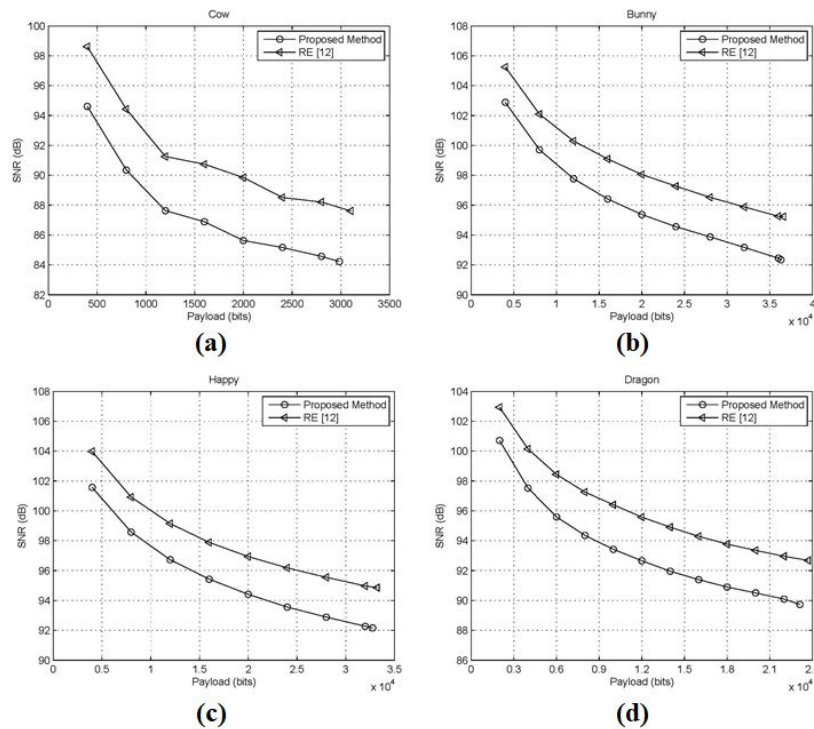


Figure 6. Distortion vs. capacity figures of the tested methods.

Table 2 shows the embedding capacities of the RE and the proposed method. It is evident that the proposed method allows for much greater embedding capacity in all four test models.

Table 2. Embedding capacity of different models.

Models	Vertices	Facets	Capacity	
			RE	Proposed
Cow	2,903	5,804	2,984	3,102
Dragon	22,998	47,794	23,118	23,704
Bunny	35,947	69,451	36,304	36,608
Happy	32,328	67,240	32,756	33,218

5. Conclusion

In this paper, the ratio expansion based watermarking method is analysed and improved. In order to find the minimum distortion for ratio expansion embedding, an expected distortion measure was modelled, and several experiments were conducted to verify the model. Based on this model, the minimum distance to the adjacent vertices is selected as a reference value, but not the minimum vertex index in [12]. Moreover, a new encoding and decoding phase is also proposed, so that more vertices are used to embed data. As a result, the embedding capacity increases. The experiments show that the proposed scheme can improve the performance on both the capacity and the distortion, and the distortion model for ratio expansion is current.

Acknowledgments

This work was supported in part by the National Nature Science Foundation of China under Grant 61701441.

References

- [1] Ai, Q. S., Liu Q., Zhou, Z. D., Yang, L., and Xie, S. Q. (2009) A new digital watermarking scheme for 3D triangular mesh models, *Signal Processing*, 89:2159-2170.
- [2] Alface, P. R., Macq, B. (2007) From 3D mesh data hiding to 3D shape blind and robust watermarking: Survey, *Transactions on Data Hiding and Multimedia Security*, 2:91-115.
- [3] Benedens, O. (1999) Geometry-based watermarking of 3D models, *IEEE Computer Graphics and Applications*, 19:46-55.
- [4] Benedens, O., Busch, C. (2000) Towards blind detection of robust watermarks in polygonal models. *Computer Graphics Forum*, 19:199-208.
- [5] Benedens, O. (2003) Robust watermarking and affine registration of 3d meshes. *Lecture Notes in Computer Science*. 2578:177-195.
- [6] Chen, T. Y., Hwang, M. S., Jan, J. K. (2009) Adaptive authentication schemes for 3D mesh models. *International Journal of Innovative Computing, Information and Control*, 5:4561-4572.
- [7] Chou, D., Jhou, C. Y., Chu, S. C. (2009) Reversible watermark for 3D vertices based on data hiding in mesh formation. *International Journal of Innovative Computing, Information and Control*, 5:1893-1901.
- [8] Garcia, F., Dugelay, J (2003) Texture-based watermarking of 3-d video objects. *IEEE Transactions on Circuits and Systems for Video Technology*, 13:853-866.
- [9] Jhou, C. Y., Pan, J. S., Chou, D. (2007) Reversible data hiding base on histogram shift for 3D vertex. In: the IEEE 2007 International Conference on Intelligent Information Hiding and Multimedia Signal Processing. Kaohsiung. pp.365-368.
- [10] Kim, H. J., Sachnev, V., Shi, Y. Q., Nam, J., Choo, H. G. (2008) A novel difference expansion transform for reversible data embedding. *IEEE Transactions on Information Forensics and Security*, 3:456-465.
- [11] Kamstra, L. H. J., Heijmans, A. M (2005) Reversible data embedding into images using wavelet techniques and sorting. *IEEE Transactions on Image Processing*, 14:2082-2090.
- [12] Ji, H. M., Yang, X. Q., Zhang, C. M., Gao, X. F. (2010) A new reversible watermarking of 3D models based on ratio expansion. In: the 3rd International Congress on Image and Signal Processing. Yantai. pp. 3899-3903.
- [13] Konstantinides, J. M., Mademlis, Daras, A., P., Mitkas P. A., Strintzis, M. G.(2009) Blind robust 3-D mesh watermarking based on oblate spheroidal harmonics. *IEEE Transactions on Multimedia*, 11:23-38.
- [14] Lin, H. S., Liao, H. M., Lu, C. S., Lin, J. C. (2005) Fragile watermarking for authenticating 3-D polygonal meshes. *IEEE Transactions on Multimedia*, 7:997-1006.
- [15] Lu, Z. M., Li, Z. (2007) High capacity reversible data hiding for 3D meshes in the PVQ domain. *Lecture Notes in Computer Science*, 5041:233-243.
- [16] Luo, H., Lu, Z. M., Pan, J. S. (2006) A reversible data hiding scheme for 3D point cloud model. In: the IEEE International Symposium on Signal Processing and Information Technology. Vancouver. pp. 863-867.
- [17] Ni, Z. Ni, Shi, Y. Q., Ansari, N., Su, W. (2006) Reversible data hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, 16:354-362.
- [18] Ohbuchi, R., Masuda, H., Aono, M. (1997) Watermarking Three Dimensional Polygonal Models. In: the ACM Multimedia. Seattle. pp. 261-272.
- [19] Ohbuchi, R., Mukaiyama, A., Takahashi, S. (2002) A frequency-domain approach to watermarking 3D shapes. *Computer Graphics Forum*, 21:373-382.
- [20] Papoulis, A. (1991) Probability, Random Variables, and Stochastic Processes. McGraw-Hill, New York.
- [21] Sachnev, V., Kim, H. J., Nam, J., Suresh, S., Shi, Y. Q. (2009) Reversible watermarking algorithm using sorting and prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, 19:989-999.

- [22] Sun, Z., Lu, Z. M., Li, Z. (2006) Reversible data hiding for 3D meshes in the PVQ-compressed domain. In: the IEEE 2006 International Conference on Intelligent Information Hiding and Multimedia Signal Processing. Pasadena. pp. 593-596.
- [23] Tai, W. L., Yeh, C.M., Chang, C. C. (2009) Reversible data hiding based on histogram modification of pixel differences. IEEE Transactions on Circuits and Systems for Video Technology, 19:906-910.
- [24] Thodi, D. M., Rodriguez, J. J. (2007) Expansion embedding techniques for reversible watermarking. IEEE Transactions on Image Processing, 16:723-730.
- [25] Tian, J. (2003) Reversible data embedding using a difference expansion. IEEE Transactions on Circuits and Systems for Video Technology, 13:989-999.
- [26] Wang, K., Lavoue, G., Denis, F., Baskurt, A. (2008) A comprehensive survey on three-dimensional mesh watermarking IEEE Transactions on Multimedia, 10:1513-1527.
- [27] Wang, Y. P., Hu, S. M. (2009) A new watermarking method for 3D models based on integral invariants. IEEE Transactions on Visualization and Computer Graphics, 15:285-294.
- [28] Wu, H. T., Yiu, M. C. (2005) A reversible data hiding approach to mesh authentication. In: the IEEE/WIC/ACM 2005 International Conference on Web Intelligence. Compiègne. pp. 774-777.
- [29] Wu, H. T., Dugelay, J. L. (2008) Reversible watermarking of 3D mesh models by prediction-error expansion. In: the IEEE 10th Workshop on Multimedia Signal Processing. Cairns. pp. 797-802.
- [30] Hou, J., Kim, D., Lee, H. (2017) Blind 3D Mesh Watermarking for 3D Printed Model by Analyzing Layering Artifact. IEEE Transactions on Information Forensics and Security, 12:2712-2725.
- [31] Yang, Y., Pintus, R., Rushmeier, H., Ivrišimtzis, I. (2017) A 3D Steganalytic Algorithm and Steganalysis-Resistant Watermarking. IEEE Transactions on Visualization and Computer Graphics, 23:1002-1013.
- [32] Dragoi, I. C., Coltuc, D. (2014) Local-prediction-based difference expansion reversible watermarking. IEEE Transactions on Image Processing, 23: 1779-1790.
- [33] Dragoi, I. C., Coltuc, D. (2015) On local prediction based reversible watermarking, IEEE Transactions on Image Processing, 24:1244-1246.
- [34] Ola, M., Zein, E., Lamiaa M., Bakrawy, E., Neveen, I. G. (2017) A robust 3D mesh watermarking algorithm utilizing fuzzy C-Means clustering, Future Computing and Informatics Journal, 2:148-156.