



Separable Reversible Data Hiding Based on Integer Mapping and Multi-MSB Prediction for Encrypted 3D Mesh Models

Zhaoxia Yin^(✉), Na Xu, Feng Wang, Lulu Cheng, and Bin Luo

Anhui Provincial Key Laboratory of Multimodal Cognitive Computation,
School of Computer Science and Technology, Anhui University, Hefei 230601, China
yinzhaoxia@ahu.edu.cn

Abstract. Extensive research has been conducted on image-based reversible data hiding in encrypted domain (RDH-ED) methods, but these methods cannot be directly applied to other cover medium, such as text, audio, video, and 3D mesh. With the widespread use of 3D mesh on the Internet, the use of 3D mesh as cover medium for RDH has gradually become a research topic. The main challenge of studying RDH based on 3D mesh is that the data structure of 3D mesh is complex and the geometric structure is irregular. In this paper, we propose a separable RDH-ED method based on integer mapping and multiple most significant bit (Multi-MSB) prediction. Firstly, all vertices of 3D mesh are divided into “embedded” set and “reference” set, and floating-point vertex values are mapped to integers. Then, sender calculates prediction error of the “embedded” set. Data hider embeds additional data by replacing the Multi-MSB of the encrypted vertex coordinates of the “embedded” set without prediction error. According to different permissions, legal recipients can obtain the original mesh, the additional data or both of them by using the proposed separable method. Experimental results prove that the proposed method outperforms state-of-the-art methods.

Keywords: Reversible data hiding · 3D mesh · Multi-MSB prediction · Encrypted domain

1 Introduction

Reversible data hiding (RDH) methods [1–6] can recover cover medium losslessly after extracting embedded data, and are widely used in military, medical, remote sensing, law enforcement and other sensitive fields. The classic RDH methods are mainly based on lossless compression [1, 2], difference expansion [3, 4] and histogram shifting [5, 6]. With the rapid development of cloud computing, the demand for privacy protection is growing. In order to store or share files safely, the original content is converted to unreadable ciphertext using encryption, and then additional data is embedded directly in the ciphertext. In some scenarios,

it is vital to restore the original content losslessly after decrypting and extracting the additional data. This privacy protection scenario triggers RDH-ED to manage ciphertext data. RDH-ED methods can be mainly classified into two categories: vacating room after encryption (VRAE) [7–9] and reserving room before encryption (RRBE) [10, 11].

Zhang et al. [7] proposed a VRAE method for the first time. The encrypted image was divided into several non-overlapping blocks, and additional data was embedded by flipping the least significant bit (LSB) of the encrypted data. With the emergence of new application requirements in different scenarios, Zhang et al. [8] proposed a separable RDH-ED method for the first time. In this method, the least significant bit (LSB) of the encrypted image was losslessly compressed by a specific matrix multiplication, and the separability of data extraction and image restoration was realized. Compared with VRAE method, RRBE method has better performance in the accuracy of data extraction and image restoration. Ma et al. [10] first proposed RRBE method, which achieved data extraction error-free and lossless image restoration, and truly achieved reversibility. This method released a part of the original image by shifting the histogram, and embedded data by replacing the LSB of the encrypted pixel. Then, Puteaux et al. [11] predicted the MSB of the pixel, and embedded additional data through bit replacement according to the position indicator map.

Image-based RDH methods have been extensively studied, but these methods cannot be directly applied to other cover medium, such as text, audio, video and 3D mesh. Due to the wide application and huge inherent capacity of 3D mesh, 3D mesh was considered as a potential covering medium for RDH. At present, the research was still in its infancy, and many scientific and technological problems need to be solved. According to the literature, the existing RDH methods based on 3D model were mainly divided into four domains: spatial domain [12–14], transform domain [15], compressed domain [16] and encrypted domain [17–19].

The method in [17] was the first work on reversible data hiding in encrypted 3D mesh. Jiang et al. [17] flipped the LSBs of each vertex to embed one bit of data. The recipient used a smoothness estimation function for data extraction and mesh recovery. In order to improve the embedding capacity, Moshin et al. [18] proposed a two-layer embedding scheme based on homomorphic encryption. The sender of the first-tier used the histogram shifting to embed the additional data, and the second-tier cloud manager used the self-blind property of homomorphic encryption to embed the additional data into the marked encrypted mesh. However, due to the large ciphertext expansion and high computational complexity of Paillier cipher system, this method [18] was not efficient in practical application. Recently, Tsai et al. [19] used spatial coding technology with embedding threshold to embed additional data, and the embedding capacity has been improved. Due to the bit error rate in data extraction, the application of this method has certain limitations. In order to make full use of the model local correlation, we embedded additional data on the x -axis, y -axis, and z -axis of the vertices, replacing the n MSB bits of each coordinate axis with n bits of additional data. Compared with [17–19], the proposed method vacated more room for additional data embedding.

In this paper, we propose a separable RDH method for encrypted 3D mesh based on integer mapping and Multi-MSB prediction. The main contributions of this paper are as follows: (1) Multi-MSB embedding strategy is adopted to obtain higher embedding capacity. (2) By making full use of the correlation of adjacent vertices in natural mesh, the recipient can recover the Multi-MSB of the “embedded” vertex by Ring-prediction, so as to achieve lossless recovery mesh. (3) The proposed method can ensure the data extraction is error-free and separable, which is of great significance to privacy protection.

In this paper, Sect. 2 introduces the proposed method. Section 3 presents the analysis of experimental results. Section 4 concludes this paper and describes the future work.

2 Proposed Method

Figure 1 shows the framework of the proposed method. Firstly, the sender divides the original 3D mesh into “embedded” set and “reference” set. The sender analyzes the vertex information with prediction errors in the “embedded” set and records it as auxiliary information. The data hider uses the Multi-MSB bit replacement strategy to embed additional data into the encrypted 3D mesh $E(M)$, and can obtain the marked encrypted mesh $E(M)w$. According to different permissions, legal recipients can obtain the original mesh, the additional data or both of them by using the proposed separable method.

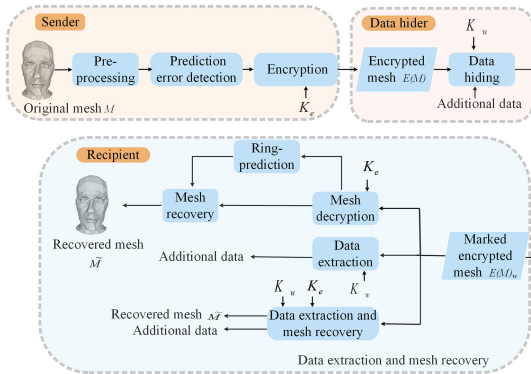


Fig. 1. Framework of the proposed method.

2.1 Pre-processing

According to the suggestion of [20], sender can perform lossy compression in different application scenarios. According to the different precision m , the corresponding integer value is between -10^m and 10^m , where $m \in [1, 33]$. Normalizing floating point coordinates $v_{i,j}$ to integer coordinates $\bar{v}_{i,j}$ as

$$\bar{v}_{i,j} = \lfloor v_{i,j} \times 10^m \rfloor, \tag{1}$$

where i is the i th vertex, $j \in \{x, y, z\}$, $v_{i,j}$ is the original set of floating point vertices and $\bar{v}_{i,j}$ is the set of integer vertices.

Recipient can convert the processed integer coordinates back to floating point coordinates by Eq. (2).

$$\hat{v}_{i,j} = \bar{v}_{i,j}/10^m, \tag{2}$$

the value of m corresponds to the bit-length l of integer coordinates as

$$l = \begin{cases} 8, & 1 \leq m \leq 2 \\ 16, & 3 \leq m \leq 4 \\ 32, & 5 \leq m \leq 9 \\ 64, & 10 \leq m \leq 33. \end{cases} \tag{3}$$

2.2 Prediction Error Detection

The “embedded” set s_e is used to embed additional data, and the “reference” set s_n is used to recover the mesh without modifying the vertices during the whole process. We traverse all the vertices contained in the face data in ascending order, and assume that $F = (f_1, f_2 \dots f_M)$ represents the face data sequence, where $f_i = (v_{i,x}, v_{i,y}, v_{i,z})$, M is the number of face data. Assuming that $f_n = (v_{n,x}, v_{n,y}, v_{n,z})$ is the next face sequence to be traversed, and both s_e and s_n are initially 0. If there is no vertex in f_n in s_e or s_n , we choose the first vertex in f_n to add $f_{n,x}$ to s_e , and add $f_{n,y}$ and $f_{n,z}$ to s_n . Figure 2 shows the close view of the Cow mesh and its vertex connections. The blue vertices in Fig. 2 are the “embedded” set vertices, the vertices marked in yellow are the “reference” vertices, and the red vertices are the vertices to be traversed.

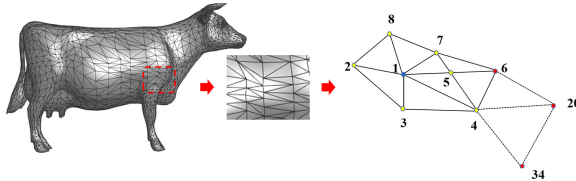


Fig. 2. The close view of Cow mesh and its vertex connection relationship. (Color figure online)

An example shows that when the m is 6, the process of selecting the maximum embedding length L without prediction error is as follows: we predict each bit of the prediction coordinates in the order of reference coordinates from MSB to LSB until a certain bit has prediction error. As shown in Fig. 3, the x coordinates of an “embedded” vertex numbered 1 has the MSB 0. The sender counts the number of 0 and 1 occurrences of the MSB of the “reference” vertex coordinates numbered 2, 3, 4, 5, 7, 8. If the number of 0s is greater than or equal to the number of 1s, the MSB of the “embedded” vertex coordinates numbered 1 is predicted to be 0. Then, the prediction of 2-MSB and 3-MSB are counted until the maximum embedding length $k1$ is found. According to Fig. 3, it can be found that the

prediction error occurred in the bit prediction when embedding length is $t1=17$. Therefore, it can be concluded that the maximum embedding length $t1$ is 16 of x coordinates when the m is 6 on the Cow mesh. The maximum embedding length of vertex coordinate x is calculated as $t1$. Similarly, we calculate the maximum embedding length of vertex coordinate y, z axes as $t2$ and $t3$. At this time, the maximum embedding length of this vertex is $\min\{t1, t2, t3\}$. In the data embedding stage, the final maximum embedding length L of the embedded vertex is the minimum embedding length of all “embedded” vertex coordinates. After the prediction error detection of the x -axis, y -axis and z -axis coordinates of the vertex coordinates, if $n \geq 1$, we call vertex numbered 1 as the “embedded” vertex without prediction error in s_e . Otherwise, the vertex index information records as auxiliary information.

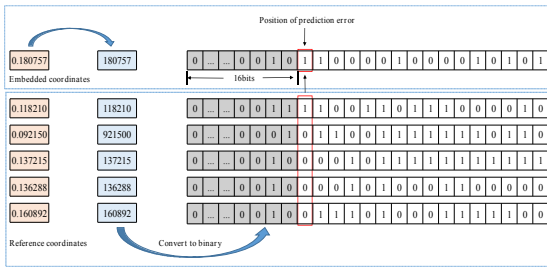


Fig. 3. An example of prediction error detection for vertex 1 of Cow mesh.

2.3 Encryption

The sender uses Eq. (4) to convert the pre-processed vertex integer coordinates into binary.

$$b_{i,j,u} = \lfloor \bar{v}_{i,j} / 2^u \rfloor \text{ mod } 2, \quad u = 0, 1 \dots l - 1, \quad (4)$$

where $\lfloor \cdot \rfloor$ is a floor function and $l \leq i \leq N$ and $j \in \{x, y, z\}$, the l of the coordinate can be obtained by Eq. (3).

The sender encrypts the bit stream of the original 3D mesh $b_{i,j,u}$ by stream cipher function generated pseudo-random bits $c_{i,j,u}$, and obtains the encrypted coordinate binary stream $e_{i,j,u}$.

$$e_{i,j,u} = b_{i,j,u} \oplus c_{i,j,u}, \quad (5)$$

where \oplus stands for exclusive OR.

The encrypted mesh can be obtained by Eq. (6)

$$E_{i,j} = \sum_{u=0}^{l-1} e_{i,j,u} \times 10^m, \quad (6)$$

where $E_{i,j}$ are the integral values of coordinates.

2.4 Data Hiding

The data hider first calculates s_e and s_n , and embeds the data into the n -MSB of the vertices in s_e without prediction error. The n -MSB of x , y , and z coordinates is replaced by n bits respectively. Finally, each vertex in s_e uses Eq. (7) to embed $3n$ bits.

$$v_{i,j}'' = s_1 \times 2^{l-1} + s_2 \times 2^{l-2} + \dots + s_n \times 2^{l-n} + v_{i,j}' \text{ mod } 2^{l-(n+1)}, \quad (7)$$

where s_k is additional data and $l \leq k \leq n$, n is embedding length and $1 \leq n \leq L$, $v_{i,j}' \in s_e$ is the vertex after pre-processing and encryption, $v_{i,j}''$ is the vertex of marked encrypted mesh.

2.5 Data Extraction and Mesh Recovery

Extraction with only Data Hiding Key. The n -MSB is extracted from the vertex coordinates of the s_e , and then the data hiding key Kw can be used to obtain the plaintext additional data.

$$s_k = v_{i,j}'' / 2^{l-k}, \quad 1 \leq k \leq n \quad (8)$$

where $v_{i,j}'' \in s_e$ is vertex of the marked encrypted mesh.

Mesh Recovery with only Encryption Key. The recipient can recover the $E(M)w$ to get the M with Encryption Key Ke . After mesh decryption and Ring-prediction, the M can be obtained.

The pseudorandom bits $c_{i,j,u}$ are generated by the encryption key Ke , and is used to perform XOR function with $e''_{i,j,u}$ to decrypt the $E(M)w$.

$$b''_{i,j,u} = e''_{i,j,u} \oplus c_{i,j,u}, \quad (9)$$

where $e''_{i,j,u}$ is the binary stream of the marked encrypted mesh, $b''_{i,j,u}$ is the binary stream of the decrypted mesh with additional data and $u = 0, 1 \dots l - 1$.

After decrypting the $E(M)w$, since the s_n has not been modified in the whole process, the coordinate value after decryption is the original coordinate value. In order to recover the coordinate values in the s_e , after decryption of the $E(M)w$, the recipient can predict the n -MSB of the embedded vertices by embedding the n -MSB of adjacent vertices around the vertices, which is called Ring-prediction.

Extraction and Mesh Recovery with both Keys. In this case, the recipient can extract the additional data and recover the original 3D mesh perfectly. Note that data extraction step needs to be performed before mesh restoration.

3 Experimental Results and Discussion

In this section, we analyze the embedding capacity and reversibility of the proposed method, and compare the performance with state-of-the-art methods [17–19]. The experimental environment is MATLAB R2018b. Four standard original meshes, Beetle, Mushroom, Mannequin, and Elephant are used to demonstrate the experimental performance. Two data sets used for performance evaluation are: The Princeton Shape Retrieval and Analysis Group¹ and The Stanford 3D Scanning Repository². In Sect. 3.1, we analyze the key indicator embedding capacity of the proposed method. In Sect. 3.2, Hausdorff distance and signal-to-noise ratio (SNR) are used to evaluate the geometric and visual quality of the proposed method, that is to evaluate reversibility. In Sect. 3.3, the performance comparison of the proposed method and the state-of-the-art methods is given. The additional data is a randomly generated 0/1 sequence.

3.1 Embedding Capacity

The embedding rate (ER) is defined as the ratio of the number of embedding bits to the number of vertices in the mesh model, that is, the number of bits per vertex (bpv). This section mainly discusses the impact on embedding rate under different m and embedding length n . Figure 4 show that when m takes a fixed value, with the increase of n , the embedding rate first increases and reaches the maximum embedding rate, and then shows a decreasing trend. Taking the Beetle as an example, when $m = 2$ and $n = 5$, the embedding rate is 3.70 bpv. When $m = 3$ and $n = 8$, the embedding rate is 7.74 bpv. When $m = 5$ and $n = 17$, the maximum embedding rate is 16.51 bpv. When $m = 6$ and $n = 14$, the embedding rate is 13.64 bpv, and the embedding rate shows a downward trend. Similarly, when $m = 5$ and $n = 15$, the maximum embedding rate of the Mushroom is 16.72 bpv. When $m = 5$ and $n = 15$, the maximum embedding rate of the Mannequin is 13.66 bpv. When $m = 5$ and $n = 18$, the maximum embedding rate of Elephant is 18.12 bpv.

3.2 Geometric and Visual Quality

For differences that cannot be distinguished by the naked eye, Hausdorff distance and SNR can be used to measure the geometric distortion. Hausdorff distance is defined as follows:

$$H(A, B) = \max(h(A, B), h(B, A)), \quad (10)$$

$$h(A, B) = \max(a \in A) \min(b \in B) \| a - b \|, \quad (11)$$

$$h(B, A) = \max(b \in B) \min(a \in A) \| b - a \|, \quad (12)$$

¹ <http://shape.cs.princeton.edu/benchmark/index.cgi>.

² <http://graphics.stanford.edu/data/3Dscanrep/>.

where $\| \cdot \|$ is the distance between point a of set A and point b of set B (such as L2), a and b are the number of elements in the set.

The signal-to-noise ratio (SNR) is defined as follows: $SNR=$

$$10 \times \lg \frac{\sum_{i=1}^N [(v_{i,x} - \bar{v}_x)^2 + (v_{i,y} - \bar{v}_y)^2 + (v_{i,z} - \bar{v}_z)^2]}{\sum_{i=1}^N [(g_{i,x} - \bar{v}_x)^2 + (g_{i,y} - \bar{v}_y)^2 + (g_{i,z} - \bar{v}_z)^2]}, \quad (13)$$

where $\bar{v}_x, \bar{v}_y, \bar{v}_z$ are the averages of the mesh coordinates, $v_{i,x}, v_{i,y}, v_{i,z}$ are the original coordinates, $g_{i,x}, g_{i,y}, g_{i,z}$ are the modified mesh coordinates, N is the number of vertices.

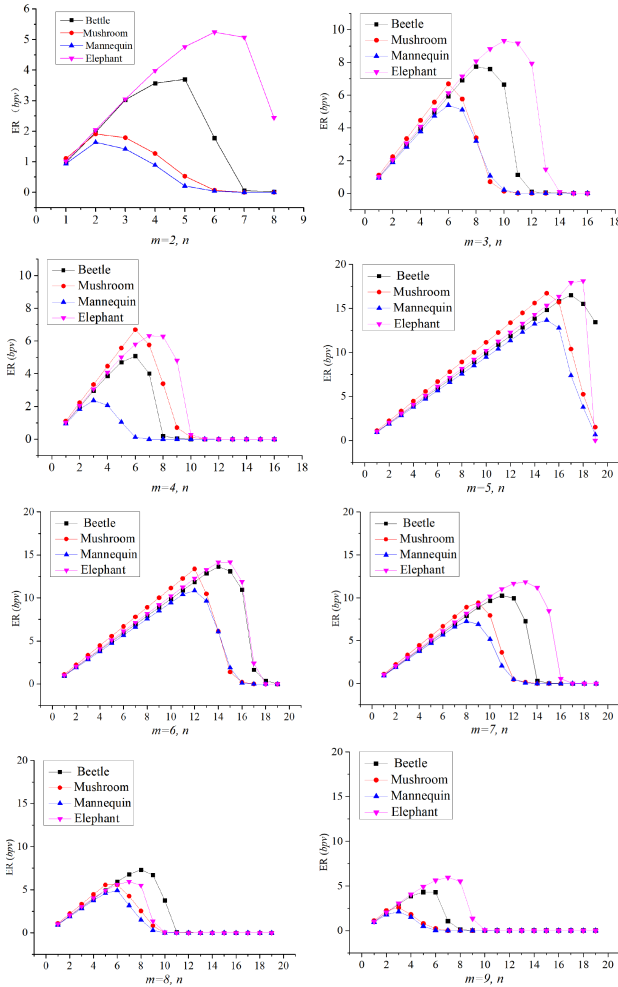


Fig. 4. The trend of embedding rate under different m and n on four test meshes.

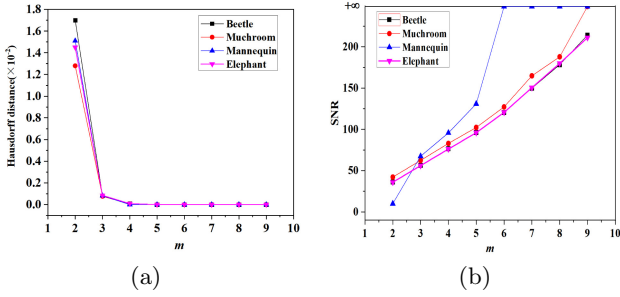


Fig. 5. The Hausdorff distance and SNR on different m under maximum embedding rate on four test meshes. (a) Hausdorff distance, (b) SNR.

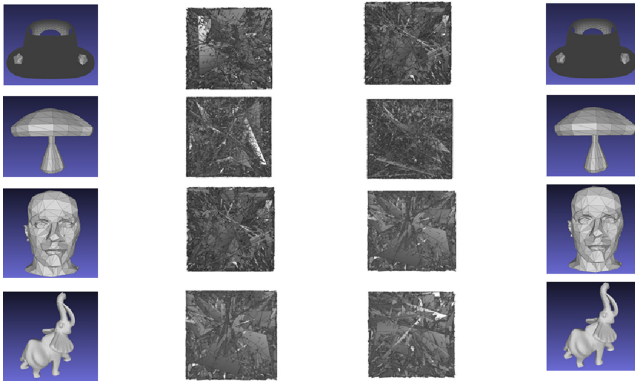


Fig. 6. Illustrative examples showing the appearance of the mesh of different stages when $m = 5$. From left to right is the original mesh, encrypted mesh, marked encrypted mesh and recovered mesh.

Figure 5(a) shows that with the increase of m , the recipient gets a higher quality recovered mesh. As shown in Fig. 5(b), as m increases, SNR gradually increases and tends to ∞ , which indicates that the recovered meshes becomes higher. Thus, the proposed method achieves reversibility by adjusting m . Figure 6 shows the visual effects of each stage when $m = 5$, including the original mesh, the encrypted mesh, the marked encrypted mesh, and the recovered mesh. Figure 6 shows that in terms of visual quality, there is no visually perceptible difference between the original mesh and the recovered mesh obtained by the proposed method, that is, the method proposed does not introduce perceptible distortion.

3.3 Performance Comparison

In this section, we compared the embedding rate of our method with methods [17–19]. Jiang et al. [17] embeds 1bit into each vertex by flipping the three

LSBs of the vertex in the data hiding stage. Since the embedding rate is limited by the mesh connectivity, the embedding rate is lower than 0.5bpv. Mohsin’s method [18] after two layers of embedding, the embedding rate reaches 6 bpv. Tsai et al. [19] uses spatial coding technology with embedding threshold to embed additional data into the vertices, and the embedding rate is about 7.68 bpv. The proposed method replaces the n MSB bits of the vertices in the “embedded” set with the n bits of the additional data, so each vertex is embedded with $3n$ bits. The proposed method makes full use of the local correlation of the model, and the embedding rate is improved. Figure 7(a) shows that the embedding rate of the proposed method on Beetle is 16.51 bpv, while the embedding rate of Jiang et al., Mohsin and Tsai et al. are 0.35 bpv, 6.00 bpv and 7.68 bpv, respectively.

In order to verify the effectiveness of the experiment, we tested the performance of embedding rate on the Princeton Shape Retrieval and Analysis Group data set. It can be seen from Fig. 7(b) that the average embedding rate of the proposed method is 14.25 bpv, while the average embedding rate of Jiang et al., Mohsin et al. and Tsai et al. method is 0.36 bpv, 6.00 bpv and 7.68 bpv, respectively. In summary, the experimental results show that the embedding rate of the proposed method is higher than that of the state-of-the-art methods [17–19].

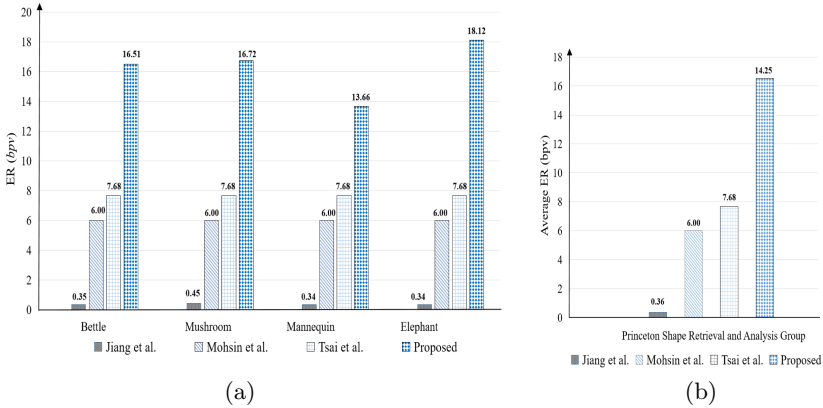


Fig. 7. Comparison of the proposed method and the state-of-the-art methods on embedding rate. (a) Maximum embedding rate on four test meshes. (b) Average embedding rate on Princeton Shape Retrieval and Analysis Group data set.

3.4 Feature Comparison

Table 1 shows the feature comparison of the proposed method and the state-of-the-art methods [17–19]. Jiang et al. [17] method and Mohsin et al. [18] method must decrypt the mesh when extracting additional data, both of which are inseparable methods. The proposed method realizes that data extraction and mesh recovery are separable. The average bit error rate of the data extracted by the method of Jiang et al. is 4.22%. In the method of Tsai et al. [19], the data extraction error is relatively large. The proposed method can extract data completely

without error. In addition, by combining the value of m and Ring-prediction, the proposed method achieves error-free in mesh recovery stage. E_d represents error-free in data extraction. E_r represents error-free in mesh recovery.

Table 1. Feature comparison of the proposed method and the state-of-the-art methods.

Methods	Features		
	Separable	E_d	E_r
Jiang et al. [17]	×	×	×
Mohsin et al. [18]	×	✓	✓
Tsai et al. [19]	✓	×	×
Proposed	✓	✓	✓

3.5 Performance Analysis on Dense Meshes

The experimental results in Table 2 show that the experiments on dense meshes show the applicability and effectiveness of the method. Taking Dragon as an example, we can find that the maximum embedding rate is 17.3890 bpv, while the Hausdorff distance is 0.0086 (10^{-3}), the SNR is 101.1375. The data shows that this method also achieves a higher embedding rate on dense meshes. The recipient obtains a higher-quality recovered mesh through the Ring-prediction.

Table 2. Performance analysis on dense meshes

Test meshes	Embedding rate (bpv)	Hausdorff distance (10^{-3})	SNR
Dragon	17.3890	0.0086	101.1375
Armadillo	14.8133	0.0004	104.1417
Happyvrip	19.6532	0.0008	100.4581

4 Conclusions

In this paper, a separable RDH method for encrypted 3D mesh based on integer mapping and Multi-MSB prediction is proposed. The proposed method achieves a balance between capacity and distortion. To get large embedding capacity while ensuring data extraction and mesh recovery separable and error-free, Multi-MSB embedding strategy is used. To obtain high-quality recovered mesh, Ring-prediction is adopted. The results prove that the proposed method improves the embedding capacity compared with the state-of-the-art methods. A major limitation is that the embedding capacity is still limited by mesh connectivity. How to design a more effective algorithm to improve the embedding capacity could be the future work.

Acknowledgment. This research work is partly supported by National Natural Science Foundation of China (61872003, 61860206004).

References

1. Fridrich, J., Goljan, M., Rui, D.: Lossless data embedding for all image formats. *SPIE Secur. Watermarking Multimed. Contents* **4675**, 572–583 (2002)
2. Celik, M.U., Sharma, G., Tekalp, A.M., Saber, E.: Lossless generalized-LSB data embedding. *IEEE Trans. Image Process.* **14**(2), 253–266 (2005)
3. Tian, J.: Reversible data embedding using a difference expansion. *IEEE Trans. Circ. Syst. Video Technol.* **13**(8), 890–896 (2003)
4. Hu, Y., Lee, H.-K., Chen, K., Li, J.: Difference expansion based reversible data hiding using two embedding directions. *IEEE Trans. Multimed.* **10**(8), 1500–1512 (2008)
5. Li, X., Zhang, W., Gui, X., Yang, B.: Efficient reversible data hiding based on multiple histograms modification. *IEEE Trans. Inf. Forensics Secur.* **10**(9), 2016–2027 (2015)
6. Wang, J., Ni, J., Zhang, X., Shi, Y.: Rate and distortion optimization for reversible data hiding using multiple histogram shifting. *IEEE Trans. Cybern.* **47**(2), 315–326 (2016)
7. Zhang, X.: Reversible data hiding in encrypted image. *Signal Process. Lett. IEEE* **18**(4), 255–258 (2011)
8. Zhang, X.: Separable reversible data hiding in encrypted image. *IEEE Trans. Inf. Forensics Secur.* **7**(2), 826–832 (2011)
9. Qian, Z., Zhang, X.: Reversible data hiding in encrypted images with distributed source encoding. *IEEE Trans. Circ. Syst. Video Technol.* **26**(4), 636–646 (2015)
10. Ma, K., Zhang, W., Zhao, X., Nenghai, Yu., Li, F.: Reversible data hiding in encrypted images by reserving room before encryption. *IEEE Trans. Inf. Forensics Secur.* **8**(3), 553–562 (2013)
11. Puteaux, P., Puech, W.: An efficient MSB prediction-based method for high-capacity reversible data hiding in encrypted images. *IEEE Trans. Inf. Forensics Secur.* **13**(7), 1670–1681 (2018)
12. Wu, H., Cheung, Y.: A reversible data hiding approach to mesh authentication. In: *IEEE/WIC/ACM International Conference on Web Intelligence, Compiegne, France* (2005)
13. Fei, P., Bo, L., Min, L.: A general region nesting based semi-fragile reversible watermarking for authenticating 3D mesh models. *IEEE Trans. Circ. Syst. Video Technol.* **PP**(99), 1 (2021)
14. Jiang, R., Zhang, W., Hou, D., Wang, H., Nenghai, Yu.: Reversible data hiding for 3D mesh models with three-dimensional prediction-error histogram modification. *Multimed. Tools Appl.* **77**(5), 5263–5280 (2018)
15. Wu, H., Cheung, Y.: A reversible data hiding approach to mesh authentication. In: *IEEE/WIC/ACM International Conference on Web Intelligence* (2005)
16. Li, L., Li, Z., Liu, S., Li, H.: Rate control for video-based point cloud compression. *IEEE Trans. Image Process.* **29**, 6237–6250 (2020)
17. Jiang, R., Zhou, H., Zhang, W., Nenghai, Yu.: Reversible data hiding in encrypted three-dimensional mesh models. *IEEE Trans. Multimed.* **20**(1), 55–67 (2017)
18. Shah, M., Zhang, W., Honggang, H., Zhou, H., Mahmood, T.: Homomorphic encryption-based reversible data hiding for 3D mesh models. *Arab. J. Sci. Eng.* **43**(12), 8145–8157 (2018)

19. Tsai, Y.: Separable reversible data hiding for encrypted three-dimensional models based on spatial subdivision and space encoding. *IEEE Trans. Multimed.* **23**, 2286–2296 (2020)
20. Deering, M.: Geometry compression. In: *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, pp. 13–20, New York, United States, September 1995